

UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE QUITO – CAMPUS SUR

CARRERA DE INGENIERÍA DE SISTEMAS

**ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
AUTOMATIZACIÓN Y CONTROL ORIENTADO A LA GENERACIÓN
DE RETENCIONES DEL ROL DE PAGOS EN AMBIENTE WEB,
CON NOTIFICACIONES SMS PARA LA ASOCIACIÓN DE
EMPLEADOS DE LA PRESIDENCIA DE LA REPÚBLICA DEL
ECUADOR DESARROLLADO CON SOFTWARE LIBRE.**

TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS

WILSON ALEJANDRO HIDALGO ESPIN

DIRECTOR: ING. WASHINGTON RAMÍREZ MSC.

Quito, Marzo 2013

DECLARACIÓN

Yo, Wilson Alejandro Hidalgo Espin, declaro bajo juramento que el presente trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Universidad Politécnica Salesiana, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normatividad institucional vigente.

Alejandro Hidalgo

CERTIFICACIÓN.

Certifico que el presente trabajo fue desarrollado por el Sr. Wilson Alejandro Hidalgo Espin bajo mi supervisión

Ing. Washington Ramírez Msc.
Director de tesis

AGRADECIMIENTOS

Agradezco a Dios por haberme dado la vida y mis primeros maestros que son mis padres, que depositaron su confianza, para llegar hacer un profesional con el fruto de su esfuerzo, sus ánimos para mirar siempre hacia delante y no desfallecer, por sus consejos, por compartir conmigo cada momento y por enseñarme que las metas se consiguen con esfuerzo, para después ver los frutos.

A mi hermano, mi enamorada Karla Panchi, mi familia y amigos que me brindaron su apoyo y fuerza para seguir adelante y no darme por vencido, y a todas las personas que me permitieron compartir un instante de sus vidas forjando la mía.

A mi Universidad, mis maestros y compañeros que con sus conocimientos supieron formar en mí una persona capaz de enfrentarse con el mundo de la mejor manera.

DEDICATORIA

Sin lugar a duda mi dedicatoria va dirigida a las personas que me brindaron su completo apoyo en las decisiones que he tomado a lo largo de mi vida, mis padres quienes han hecho posible cumplir mis metas, ellos que junto a mi hermano Daniel son el motor que impulsa mi vida.

A mis abuelitos, primos y primas que son la fuerza necesaria para seguir adelante día tras día, con el hecho de saber que están a mi lado, me permiten sentir que la vida es bella.

A mi familia, amigos y amigas por sus consejos, preocupación, ánimos y por estar pendiente de mis caídas y triunfos.

CONTENIDO

1	DATOS INFORMATIVOS.....	1
1.1	PLANTEAMIENTO DEL PROBLEMA	1
1.2	OBJETIVOS DEL PROYECTO	2
1.2.1	GENERAL	2
1.2.2	ESPECÍFICOS	3
1.3	JUSTIFICACIÓN DEL PROYECTO.....	4
1.4	ALCANCE DEL PROYECTO	5
2	FUNDAMENTACIÓN TEÓRICA.....	10
2.1	METODOLOGÍAS.....	10
2.1.1	ANÁLISIS COMPARATIVO DE METODOLOGÍAS DE PROGRAMACIÓN.	10
2.1.2	RESULTADO DE METODOLOGÍA.	11
2.1.3	METODOLOGÍA DE DESARROLLO XP	11
2.1.4	FASES DE LA METODOLOGÍA XP	12
2.1.5	PLANIFICACIÓN.....	12
2.1.6	DISEÑO.....	14
2.1.7	DESARROLLO.....	15
2.1.8	PRUEBAS	17
2.2	MODELO DE PROCESO DE SOFTWARE.....	18
2.2.1	UML.....	18
2.2.2	MODELAMIENTO DE CLASES.....	19
2.2.3	CASOS DE USO.....	21
2.2.4	DIAGRAMA DE INTERACCIÓN.....	22
2.3	SOFTWARE LIBRE	23
2.3.1	LIBERTADES DEL SOFTWARE LIBRE	24
2.3.2	TIPOS DE LICENCIAS	24

2.3.3	VENTAJAS.....	25
2.3.4	DESVENTAJAS.	26
2.3.5	SOFTWARE LIBRE EN ECUADOR	26
2.4	SISTEMAS OPERATIVOS PARA SERVIDORES.....	27
2.4.1	CENTOS.....	28
2.4.2	ANÁLISIS COMPARATIVO ENTRE SERVIDORES.....	28
2.4.3	RESULTADO DEL ANALISIS	30
2.5	LENGUAJE DE PROGRAMACIÓN	30
2.5.1	JAVA	31
2.5.2	JSF (JAVA SERVER FACES)	31
2.5.3	PERSISTENCIAS	33
2.5.4	SERVIDOR DE APLICACIONES	36
2.6	GESTOR DE BASE DE DATOS	41
2.6.1	POSTGRESQL.....	42
2.6.2	ANÁLISIS COMPARATIVO DE GESTORES DE BASES DE DATOS ...	43
2.6.3	RESULTADO DE GESTOR DE BASE DE DATOS	44
2.7	SERVICIOS.....	44
2.7.1	WEB SERVICES.....	44
2.7.2	APLICACIÓN MOVIL.....	47
2.7.3	SERVICIO SMS	50
2.8	CAPACIDAD DE ENDEUDAMIENTO SEGÚN LOSEP.	51
3	ANÁLISIS Y DISEÑO	52
3.1	PLANIFICACIÓN XP	52
3.1.1	HISTORIAS DE USUARIOS	52
3.1.2	TARJETAS CRC	62
3.1.3	RESTRICCIONES	66

3.1.4	REQUERIMIENTOS DEL SISTEMA	68
3.2	DISEÑO DEL SISTEMA	69
3.2.1	DIAGRAMAS DE CASOS DE USO	69
3.2.2	DIAGRAMAS DE ESTADO	87
3.2.3	DIAGRAMAS DE SECUENCIA	93
3.2.4	DIAGRAMA NAVEGACIONAL	98
3.2.5	DIAGRAMA DE CLASES	103
3.2.6	DIAGRAMA DE BASE DE DATOS	104
3.2.7	DICCIONARIO DE DATOS	106
3.2.8	DISEÑO DE LA INTERFAZ Y NAVEGACIÓN	114
3.2.9	DISEÑO DE SOFTWARE	117
4	IMPLEMENTACIÓN Y PRUEBAS	119
4.1	CÓDIGO FUENTE	119
4.1.1	PLANTILLA DE LA CAPA PRESENTACIÓN.	119
4.1.2	BEANS.....	121
4.1.3	MANEJO DE WEB SERVICE	124
4.1.4	CAPA ANDROID	126
4.2	MANUALES	127
4.3	PRUEBAS	127
4.3.1	PRUEBAS DE CAJA NEGRA	128
4.3.2	PRUEBAS DE CAJA BLANCA	145
5	CONCLUSIONES Y RECOMENDACIONES	163
5.1	CONCLUSIONES	163
5.2	RECOMENDACIONES	165
	GLOSARIO DE TÉRMINOS.....	167
	BIBLIOGRAFÍA.....	172

CONTENIDO DE FIGURAS

Figura 2-1: Extreme Programming.	12
Figura 2-2: Representación Gráfica de una Clase en UML	19
Figura 2-3: Arquitectura Hibernate.	35
Figura 2-4: Ejemplo Java Bean.	35
Figura 2-5: Ejemplo HQL.	36
Figura 2-6: Servidor de Aplicaciones	37
Figura 2-7: Arquitectura MVC	38
Figura 2-8 Arquitectura WEB SERVICE.....	46
Figura 3-1 Diagrama Inicio de Sesión de Usuarios.....	69
Figura 3-2 Diagrama Creación de Usuarios	70
Figura 3-3 Diagrama Ingreso de Casas Comerciales.....	71
Figura 3-4 Registro de Bancos	72
Figura 3-5 Diagrama Registro de Cuenta Bancaria	73
Figura 3-6 Diagrama Ingreso de Cheques.	74
Figura 3-7 Diagrama Registro de Facturas.	75
Figura 3-8 Diagrama Ingreso de Préstamos.	76
Figura 3-9 Diagrama Ingreso de Aportes.....	77
Figura 3-10 Diagrama Generación de Retención de Préstamos.	78
Figura 3-11 Diagrama Buscar Retenciones de Préstamos.....	80
Figura 3-12 Diagrama Generación de Retención de Aportes.	81
Figura 3-13 Diagrama Buscar Retenciones de Aportes.	83
Figura 3-14 Diagrama Agregar Retención de Factura.....	84
Figura 3-15 Diagrama Generación de Retenciones Totales.	85
Figura 3-16 Diagrama de Inicio de Sesión.	87
Figura 3-17 Diagrama Registro de Facturas.	88
Figura 3-18 Diagrama Registro Cuenta Bancaria.	89
Figura 3-19 Diagrama Generar Retención Aportes.....	90
Figura 3-20 Agregar Retención Facturas.....	91

Figura 3-21 Diagrama Generar Retención Préstamos.	92
Figura 3-22 Diagrama Generación Retención Total.	93
Figura 3-23 Diagrama Generación Inicio de Sesión.	94
Figura 3-24 Diagrama Generación Registro de Facturas.	95
Figura 3-25 Diagrama Registro de Cuentas Bancarias.	96
Figura 3-26 Diagrama Generación Retención Total por Mes.	97
Figura 3-27 Diagrama Navegacional Administrador_1	98
Figura 3-28 Diagrama Navegacional Administrador_2	99
Figura 3-29 Diagrama Navegacional Secretaria_1	100
Figura 3-30 Diagrama Navegacional Secretaria_2	101
Figura 3-31 Diagrama Navegacional Dispositivo Móvil	101
Figura 3-32 Diagrama Navegacional Asociado.....	102
Figura 3-33 Diagrama de Clases.....	103
Figura 3-34 Modelo Lógico de Bdd	104
Figura 3-35 Modelo Físico de Bdd	105
Figura 3-36 Diseño de Pantalla	114
Figura 3-37 Sitio Web Encabezado	115
Figura 3-38 Sitio Web Menú.....	115
Figura 3-39 Sitio WebContenido	116
Figura 3-40 Sitio Web Pie de Página.....	116
Figura 3-41 Sitio Web Home	117
Figura 3-42 AsoPresi Modelo Vista Controlador.....	118
 Figura 4-1 Código de Página Plantilla	 121
Figura 4-2 Código Blogin.Java.....	124
Figura 4-3 Código Web Service: Servicemensual.Java	126
Figura 4-4 Código Android Prestamo_Activity.Xml	127

CONTENIDO DE TABLAS

Tabla 2-1: Metodologías de Desarrollo de Software.....	11
Tabla 2-2: Elementos Necesarios para un Diagrama de Casos de Uso.	22
Tabla 2-3: Componentes de un Diagrama de Interacción.	23
Tabla 2-4: Comparación de Software Propietario y Libre	24
Tabla 2-5 Requerimientos de Instalación del Servidor	29
Tabla 2-6: Análisis de Sistemas Operativos	30
Tabla 2-7: Lenguajes de Programación más Utilizados	31
Tabla 2-8: Características de Tecnologías Persistentes.	34
Tabla 2-9: Servidores de Aplicaciones	40
Tabla 2-10 Gestores de Base de Datos más Conocidos	43
Tabla 2-11: Análisisde Software para Móviles	48
Tabla 3-1 Historia de Usuario: Requerimientos Iniciales.....	52
Tabla 3-2 Historia de Usuario: Tipos de Perfiles	53
Tabla 3-3 Historia de Usuario: Base de Datos	53
Tabla 3-4 Historia de Usuario: Desarrollo de Inicio de Sesión.....	54
Tabla 3-5 Historia de Usuario: Gestión de Usuarios.....	54
Tabla 3-6 Historia de Usuario: Menús	55
Tabla 3-7 Historia de Usuario: Páginas Estándares Listar	55
Tabla 3-8 Historia de Usuario: Páginas Estándares Editar.....	56
Tabla 3-9 Historia de Usuario: Páginas Estándares Crear Nuevo	56
Tabla 3-10 Historia de Usuario: Opción Retenciones	57
Tabla 3-11 Historia de Usuario: Retenciones Factura	58
Tabla 3-12 Historia de Usuario: Retenciones Totales	58
Tabla 3-13 Historia de Usuario: Reportes.....	59
Tabla 3-14 Historia de Usuario: Archivo Retenciones	59
Tabla 3-15 Historia de Usuario: Web Service	60
Tabla 3-16 Historia de Usuario: Corrección Listar Facturas	60
Tabla 3-17 Historia de Usuario: Corrección Egresos.....	60

Tabla 3-18 Historia de Usuario: Envío de sms	61
Tabla 3-19 Historia de Usuario: Menú Asociado1	61
Tabla 3-20 Historia de Usuario: Menú Asociado2	62
Tabla 3-21 Tarjeta Crc: Sesiones.....	62
Tabla 3-22 Tarjeta Crc: Usuarios	62
Tabla 3-23 Tarjeta Crc: Bancos.....	63
Tabla 3-24 Tarjeta Crc: Cheques	63
Tabla 3-25 Tarjeta Crc: Casas Comerciales	63
Tabla 3-26 Tarjeta Crc: Facturas.....	63
Tabla 3-27 Tarjeta Crc: Préstamos	64
Tabla 3-28 Tarjeta Crc: Aportes	64
Tabla 3-29 Tarjeta Crc: Prestamos	64
Tabla 3-30 Tarjeta Crc: Aportes	64
Tabla 3-31 Tarjeta Crc: Facturas.....	65
Tabla 3-32 Tarjeta Crc: Retenciones Totales.....	65
Tabla 3-33 Tarjeta Crc: Exportar Archivo.....	65
Tabla 3-34 Tarjeta Crc: Web Service	65
Tabla 3-35 Tarjeta Crc: Cuentas Asociación.....	65
Tabla 3-36 Caso de Uso Inicio de Sesión de Usuarios.	70
Tabla 3-37 Caso de Uso Creación de Usuarios.	71
Tabla 3-38 Caso de Uso Ingreso de Casas Comerciales.	72
Tabla 3-39 Caso de Uso Registro de Bancos	72
Tabla 3-40 Caso de Uso Registro de Cuenta Bancaria	74
Tabla 3-41 Caso de Uso Ingreso de Cheques.....	75
Tabla 3-42 Caso de Uso Registro de Facturas.....	76
Tabla 3-43 Caso de Uso Ingreso de Préstamos.....	77
Tabla 3-44 Caso de Uso Ingreso de Aportes.....	77
Tabla 3-45 Caso de Uso Generación de Retención de Préstamos.....	78
Tabla 3-46 Caso de Uso Agregar Nueva Retención de Préstamos.....	79
Tabla 3-47 Caso de Uso Eliminar o Actualizar Retención de Préstamos.....	79
Tabla 3-48 Caso de Uso Buscar Retenciones de Préstamos.....	80

Tabla 3-49 Caso de Uso Generación de Retención de Aportes.	82
Tabla 3-50 Caso de Uso Agregar Nueva Retención de Aportes.	82
Tabla 3-51 Caso de Uso Eliminar Retención de Aportes para el Mes en Curso.	82
Tabla 3-52 Caso de Uso Buscar Retenciones de Aportes.	83
Tabla 3-53 Caso de Uso Agregar Retención de Factura.	85
Tabla 3-54 Caso de Uso Generación de Retenciones Totales.	86
Tabla 3-55 Caso de Uso Agregar Retención Total.	86
Tabla 3-56 Caso de Uso Eliminar Retención Total.	87
Tabla 3-57: Detalle Base de Datos	106
Tabla 3-58: Descripción General de las Bases de Datos	106
Tabla 3-59: Descripción Tabla Tb_Aportes	107
Tabla 3-60: Descripción Tabla Tb_Banco	107
Tabla 3-61: Descripción Tabla Tb_Casa_Comercial	108
Tabla 3-62: Descripción Tabla Tb_Cheque	108
Tabla 3-63: Descripción Tabla Tb_Ctas_Pagar	109
Tabla 3-64: Descripción Tabla Tb_Cuenta	109
Tabla 3-65: Descripción Tabla Tb_Facturas_X_Pagar.....	110
Tabla 3-66: Descripción Tabla Tb_Modulo	110
Tabla 3-67: Descripción Tabla Tb_Perfil	110
Tabla 3-68: Descripción Tabla Tb_Prestamo	111
Tabla 3-69: Descripción Tabla Tb_Retencion_Prestamo	111
Tabla 3-70: Descripción Tabla Tb_Retencion_Aportes	112
Tabla 3-71: Descripción Tabla Tb_Retencion_Fac	112
Tabla 3-72: Descripción Tabla Tb_Retencion_Total	113
Tabla 3-73: Descripción Tabla Tb_Usuario	114
Tabla 4-1 Prueba de Caja Negra: Iniciar Sesión, Datos Incorrectos	129
Tabla 4-2 Prueba de Caja Negra: Iniciar Sesión, Datos Correctos	130
Tabla 4-3 Prueba de Caja Negra: Ingreso de Usuarios, Datos Incorrectos	131
Tabla 4-4 Prueba de Caja Negra: Ingreso de Usuarios, Datos Correctos	132
Tabla 4-5 Prueba de Caja Negra: Ingreso de Cheques, Datos Incorrectos	132
Tabla 4-6 Prueba de Caja Negra: Ingreso de Cheques, Datos Correctos	133

Tabla 4-7 Prueba de Caja Negra: Ingreso de Casa Comercial, Datos Incorrectos ...	134
Tabla 4-8 Prueba de Caja Negra: Ingreso de Casa Comercial, Datos Correctos	134
Tabla 4-9 Prueba de Caja Negra: Ingreso de Factura, Datos Incorrectos	135
Tabla 4-10 Prueba de Caja Negra: Ingreso de Factura, Datos Correctos	136
Tabla 4-11 Prueba de Caja Negra: Ingreso de Préstamo, Datos Incorrectos	136
Tabla 4-12 Prueba de Caja Negra: Ingreso de Préstamo, Datos Correctos	137
Tabla 4-13 Prueba de Caja Negra: Ingreso de Aportes, Datos Incorrectos	137
Tabla 4-14 Prueba de Caja Negra: Ingreso de Aportes, Datos Correctos	138
Tabla 4-15 Prueba de Caja Negra: Retenciones de Aportes, Datos Incorrectos	139
Tabla 4-16 Prueba de Caja Negra: Retenciones de Aportes, Datos Correctos	140
Tabla 4-17 Prueba de Caja Negra: Retenciones de Préstamos, Datos Incorrectos .	141
Tabla 4-18 Prueba de Caja Negra: Retenciones de Préstamos, Datos Correctos	142
Tabla 4-19 Prueba de Caja Negra: Retenciones de Facturas, Datos Incorrectos	143
Tabla 4-20 Prueba de Caja Negra: Retenciones de Facturas, Datos Correctos	144
Tabla 4-21 Prueba de Caja Negra: Retenciones Totales, Datos Incorrectos	144
Tabla 4-22 Prueba de Caja Negra: Retenciones Totales, Datos Correctos	145
Tabla 4-23 Prueba de Caja Blanca: Validar Usuario y Contraseña	147
Tabla 4-24 Prueba de Caja Blanca: Establecer Usuario de Sesión	148
Tabla 4-25 Prueba de Caja Blanca: Consultar Préstamos por Usuario Ingresado	149
Tabla 4-26 Prueba de Caja Blanca: Ingreso de Sueldo	149
Tabla 4-27 Prueba de Caja Blanca: Validar Cédula	150
Tabla 4-28 Prueba de Caja Blanca: Consultar Cuota de Préstamo	150
Tabla 4-29 Prueba de Caja Blanca: Ingresar Código de Clave Primaria	151
Tabla 4-30 Prueba de Caja Blanca: Guardar Aporte	152
Tabla 4-31 Prueba de Caja Blanca: Modificar Retención de Aporte	152
Tabla 4-32 Prueba de Caja Blanca: Cheque no Relacionado	153
Tabla 4-33 Prueba de Caja Blanca: Facturas Recientes	154
Tabla 4-34 Prueba de Caja Blanca: Mostrar un Préstamo	154
Tabla 4-35 Prueba de Caja Blanca: Generar Retenciones de Aportes	155
Tabla 4-36 Prueba de Caja Blanca: Eliminar Retenciones de Aportes	156
Tabla 4-37 Prueba de Caja Blanca: Eliminar una Retención de Aportes	157

Tabla 4-38 Prueba de Caja Blanca: Editar o Eliminar Retenciones	158
Tabla 4-39 Prueba de Caja Blanca: Generar Retenciones de Préstamos	159
Tabla 4-40 Prueba de Caja Blanca: Generar Retenciones Totales	160
Tabla 4-41 Prueba de Caja Blanca: Generar Todas las Retenciones	161
Tabla 4-42 Prueba de Caja Blanca: Generar Retenciones ya Generadas	162

CONTENIDO DE ANEXOS

ANEXO A. CAPACIDAD DE ENDEUDAMIENTO SEGÚN LOSEP	174
ANEXO B. MANUAL DE USUARIO.....	174
ANEXO C. MANUAL TÉCNICO	174

RESUMEN

El proyecto de titulación se basa en la automatización y control de los ingresos y egresos que tienen a su cargo la Asociación de Empleados de la Presidencia de la República del Ecuador.

La puesta en marcha del software: Asociación de la Presidencia ASOPRESI, dará a conocer de manera detallada las retenciones efectuadas por los directivos, sirviendo de guía para mejorar los procesos realizados, ejecutando de manera ordenada el flujo de información con la que se trabaja. Con esto se garantizará credibilidad y confianza en las actividades para el beneficio de sus socios.

Como resultado final se tendrá un sistema en el que cada asociado acceda a la Web con su respectiva identificación, además se implementará a una aplicación móvil que estará instalada en dispositivos Android y el envío de mensajes de texto, así el asociado contará con varias herramientas a la mano y de fácil uso, permitiendo conocer en detalle el valor total de su retención.

INTRODUCCIÓN

El desarrollo del presente proyecto tiene como finalidad ayudar a la asociación de empleados de la Presidencia de la República del Ecuador, automatizando la información de ingresos y egresos de cada miembro adecuadamente, en dispositivos tecnológicos que se ajusten con las últimas tendencias que existe en el mercado comercial, principalmente que cuenten con internet como una computadora, un teléfono móvil inteligente y servicio de mensajería celular instantánea. Estos aparatos computacionales permitirán acceder, almacenar y devolver la retención mensual, que serán de conocimiento personal de cada socio, la cual será la suma de los gastos que tiene el asociado con diferentes casas comerciales, aportes propios de la asociación y préstamos,

En el capítulo uno, se describe de una forma corta y clara el planteamiento del problema, los objetivos del proyecto, las herramientas que se utilizarán y la solución del mismo definiendo los alcances y la metodología que se usará.

En el capítulo dos, se plantea la fundamentación teórica de la metodología, los procesos que tiene el software, una breve síntesis de los lenguajes de programación que se usa así como el gestor de base de datos y otros servicios necesarios para la puesta en marcha del proyecto.

En el capítulo tres, se usa las fases de la metodología XP aplicadas al proyecto y el diseño del sistema como los diagramas de casos de uso, de base de datos, etc.

En el capítulo cuatro, es la puesta en producción del software así como las pruebas, manuales y requerimientos necesarios para el funcionamiento correcto de la aplicación.

CAPITULO I

1 DATOS INFORMATIVOS.

En este capítulo se detalla la problemática, objetivos, justificación y alcance que se vayan a desarrollar través del tema de titulación, proporcionando las bases para analizar un sistema de automatización y control orientado a las retenciones del rol de pagos en ambiente Web.

1.1 PLANTEAMIENTO DEL PROBLEMA

La Asociación de empleados de la Presidencia de la República del Ecuador es un ente sin fines de lucro y autónomo, que busca la participación desinteresada de sus socios, puesto que considera que es su deber y obligación apoyar y velar por el bienestar de los empleados, mediante el buen funcionamiento de esta asociación, la cual representa a los Servidores de la Presidencia de la República del Ecuador.

En la actualidad la asociación se encarga de llevar los registros de los movimientos salariales de cada uno de los socios, almacenados de forma manual en hojas sueltas o en hojas electrónicas, manejando ingresos y egresos con una metodología poco sistematizada, lo que resulta ineficiente generando pérdida de información inestimable, principalmente por las personas encargadas de administrarla, ya sea por no registrar los pagos a una empresa afiliada, no descontar o descontar varias veces, no coincidir con los gastos de la persona, pérdida de registros de pagos, inconsistencias y duplicidad. Dado que el proceso es manual hay el riesgo de que se pueda equivocar por la cantidad de papeles que maneja, a la vez pérdida de documentación valiosa, correspondiente a los detalles de los descuentos para el rol de pagos que se le aplicará a cada uno de sus socios.

Dichos problemas generan descontrol en la administración de la información, provocando pérdida de tiempo al retrasar y sobrecargar el trabajo cada fin de mes para las personas que son responsables de entregar los reportes de pagos, lo que afecta la credibilidad de la asociación por el desconocimiento que se produce hacia los socios sobre los valores y detalles exactos que se les descuenta, y a la vez inconformidad y retraso, puesto que deben acercarse a la asociación mes a mes para averiguar sus retenciones.

Los asociados y sus principales representantes, no cuentan con información detallada que ayude a saber cómo se están efectuando los convenios con las empresas comercializadoras que prestan sus servicios, así como los gastos que se generan tanto de ingresos como de egresos, imposibilitando un seguimiento de compras, puesto que no existe una administración de inventarios, y los reportes se manejan manualmente, por lo que se generan errores significativos con las empresas afiliadas.

La parte administrativa se encarga de pedir información de la capacidad de endeudamiento de los asociados cada cierto tiempo a la dirección financiera de la presidencia, esta información también entregada en papel se envía al representante de la asociación para analizar si se puede respaldar una compra o conceder un préstamo.

1.2 OBJETIVOS DEL PROYECTO

1.2.1 GENERAL

Analizar, diseñar e implementar un sistema de automatización y control orientado a la generación de retenciones del rol de pagos en ambiente Web, con notificaciones Sms para la Asociación de Empleados de la Presidencia de la República del Ecuador desarrollado con software libre.

1.2.2 ESPECÍFICOS

- Investigar las necesidades de información requeridas por los usuarios.
- Proponer una herramienta online que permita mantener fácilmente actualizados los contenidos del espacio Web sin necesidad de conocimientos informáticos.
- Crear una aplicación funcional para cualquier dispositivo móvil, teléfono celular y computador.
- Realizar una propuesta sobre un portal Web para que los empleados puedan tener conocimiento de los eventos que se dan en la asociación y a la vez puedan interactuar entre los miembros del comité por alguna inquietud.
- Diseñar un sistema que permita gestionar los ingresos y egresos de la asociación, a la vez generar reportes y gráficos estadísticos según las necesidades del personal administrativo.
- Implementar un servidor de correo electrónico, servidor ftp y servidor de aplicación.
- Dar un servicio de valor añadido a los asociados, en el espacio Web que permita interrelación entre los mismos, mediante sesiones de chat.
- Gestionar fácilmente la Base de Datos de los asociados.
- Tomar información de la Presidencia referente a las retenciones fijas de los asociados mediante Web Service.
- Calcular la capacidad de endeudamiento que tiene cada asociado en tiempo real.

1.3 JUSTIFICACIÓN DEL PROYECTO

En la actualidad esta organización utiliza un proceso poco eficiente para realizar las diferentes actividades de administración y control de información, estos procesos deberían ser automatizados para lograr la optimización de recursos y mejorar la disponibilidad de información importante.

Buscando la satisfacción de sus socios y asociados, se desea crear un ente de comunicación más formal y constantemente actualizado, de los ingresos y egresos que genera la asociación, además de contar con un respaldo más amplio de información tanto para los asociados como para el público en general, y la posibilidad de crear nuevos nexos con empresas que estén interesadas en prestar sus servicios.

Se busca desarrollar un sistema que pueda gestionar las retenciones mensuales que efectúa la parte administrativa, mediante un software que sea capaz de tener organizada la información de cada asociado, correspondiente al salario mensual y a los descuentos que se les debita por los pagos acumulados que tienen pendiente para el mes en curso, detallando los pagos pendientes y si existe la posibilidad de endeudamiento con alguna otra casa comercial. Además permite determinar si el asociado tiene capacidad para un préstamo exprés, el cual consiste en un préstamo que administra la asociación en donde es posible realizar el pago en cómodas cuotas hasta por tres meses dependiendo del monto del préstamo y según la LOSEP (Ley Orgánica del Servidor Público).

Mediante una aplicación web, se publicarán noticias internas como externas a la asociación, logrando también incentivar el uso de correo interno para conocer temas que afectan a cada uno de los asociados.

Por medio de un dispositivo móvil cada asociado podrá conocer los gastos que serán debitados mensualmente en su rol de pagos, a través de un SMS que le llegará a su número telefónico, de igual forma podrá acceder desde su celular mediante una aplicación desarrollada para Android, la que contendrá el detalle de retenciones correspondientes al mes en curso, teniendo la información a la mano,

esto evitará el desconocimiento que se genera cada fin de mes por no llevar información verídica de las retenciones de las cuales se encarga la asociación, de esta manera tener un sistema robusto y confiable las 24 horas del día.

1.4 ALCANCE DEL PROYECTO

La necesidad que tienen las empresas por mejorar la calidad de su servicio mediante nuevas herramientas tecnológicas, hacen que resulte útil y factible implementar un sistema que permita el manejo de información.

Se propone crear un sitio Web que formalizará y hará conocer las noticias que ocurren tanto dentro como fuera de la asociación, además contendrá una intranet que permitirá a los usuarios conocer principalmente las retenciones de pagos y su cuota a futuro que se va aportar o que será debitada.

Es por esto que se desarrollará un sistema escalable y multiplataforma, mediante software libre que utilizará las siguientes herramientas.

- Java (Lenguaje de Programación)
- PostgreSQL (Gestor de Base de Datos)
- CentOS. (Sistema Operativo)

El servidor contendrá los siguientes servicios:

- GlassFish.
- PostgreSQL.
- WebMail.
- Servidor FTP.
- Chat Interno.

El desarrollo de software se lo realizará utilizando herramientas de software libre, es por esto que se han analizado lenguajes que se acoplen con las necesidades del proyecto en cuanto a costos de licencias y software propietario, se ha elegido

CentOS como sistema operativo por su robustez y administración de servicios a nivel de servidor que se pueden gestionar, este producto se libera bajo los términos de la licencia pública general de GNU.

Como lenguaje de programación se contará con JAVA ya que trabaja muy bien con software no propietario y su eficiencia en el entorno Web es reconocida a un alto nivel, incluyendo el uso de JSF que está orientado a aplicaciones en Internet permitirá realizar un trabajo eficiente con el uso de nuevas tecnologías.

Para la gestión de base de datos se utilizará PostgreSQL que se desarrolla con software libre, la cual permite una administración y acoplamiento con las herramientas antes mencionadas para el desarrollo de la aplicación que se requiere.

Los módulos de software que se desarrollarán se los detalla a continuación los cuales cubrirán las principales necesidades que requiere la asociación para un correcto manejo de la información.

Módulo Administrador.

El módulo administrador realiza las siguientes actividades:

- Ingresar al sistema como súper usuario, el cual podrá tener información resumida de la información que se lleva en la asociación.
- Enviar e-mails al personal administrativo, de los estados mensuales de ingresos y egresos.
- Administrar el sistema.
- Ingresar, modificar y eliminar asociados.
- Ingresar, modificar y eliminar cuentas y contraseñas.
- Ingresar, modificar y eliminar cuentas de correo y contraseñas.
- Administrar el portal Web.
- Cálculo de la capacidad de endeudamiento según LOSEP.

- Envío y recepción de archivo XML mensual con el uso de un Web Service hacia la Presidencia de la República.
- Gestionar Usuarios, habilitando o deshabilitando los módulos a los que se puede acceder dependiendo el usuario.

Módulo Secretaria.

Al ingresar como perfil Secretaria.

Este módulo pretende ser el apoyo que tiene la persona que realiza los registros en papel, para que se puedan automatizar los procesos.

La información que se manejará a nivel de asociados es:

- Detalle de los movimientos.
- Factibilidad que tiene para que se pueda otorgar un crédito con los distintos proveedores con que se cuenta.
- Verificación de la capacidad de endeudamiento.
- Ingreso de gastos.
- Detalle de egresos mensual.
- Envío de información al correo y al celular.

Para Todos los Asociados:

- Descuentos generales dependiendo si tiene o no nombramiento
- Ingresos de nuevos proveedores.
- Afiliación con los proveedores.
- Eliminar relación con proveedores.
- Registros de ingresos y egresos.
- Ingreso de facturas pagadas.
- Pagos mediante cheques a proveedores y préstamos de asociados.
- Generación de archivo plano, el cual detalla egresos por asociado.
- Envío de Reportes vía mail.

- Detalle de descuentos que se realizan mensualmente.
- Detalle de cheques emitidos.
- Detalle de facturas recibidas.
- Cuadro de pastel por proveedores.
- Total de ingresos que maneja la asociación.
- Ingresos y Egresos anuales.

Módulo Asociado.

Al ingresar con perfil Asociado:

Ofrece información detallada mensual de cada socio referente con los gastos que son retenidos en el mes correspondiente y en meses anteriores.

Los pagos que tienen pendientes, el número de cuotas canceladas y los pagos por cancelar, indicando el número de meses que se le descontará, además de un sistema de chat interno que servirá para la comunicación inmediata de los usuarios de la página Web.

Aplicación Móvil.

En base a las necesidades que se presentarán a futuro, se implementará una aplicación para dispositivos Android para que los usuarios puedan acceder a la página desde sus teléfonos móviles o tablets y consultar el movimiento de sus pagos.

- Acceso al portal de la asociación.
- Ingreso mediante usuario y contraseña.
- Consulta de retenciones del mes actual y de meses anteriores.

Todos estos módulos tendrán un manual de usuario explicando paso a paso como se debe manejar el sistema.

El desarrollo del sistema no contempla el uso que tenga por parte de la Presidencia, puesto que la información enviada y recibida es de mucha importancia en el traspaso de datos de lado y lado, desde la Presidencia hacia la asociación y viceversa, esto por esto que ayudará significativamente el uso de Web Service que contendrá información en un archivo XML y ya no manual como se venía trabajando con respecto a las retenciones mensuales.

De igual manera no se contará con acceso a las bases de datos de la presidencia ya que es una empresa totalmente externa.

El sistema no realizará los pagos a proveedores por cheque ni por transacciones on-line el sistema emitirá reportes de lo que se ha cancelado.

CAPITULO II

2 FUNDAMENTACIÓN TEÓRICA

2.1 METODOLOGÍAS.

La metodología es un conjunto de herramientas, procesos y técnicas, siendo un soporte documental, y de gran ayuda a la hora de realizar un nuevo software. Para el desarrollo de este proyecto se analizarán varias de las metodologías usadas con resultados favorables y luego se determinará cuál de ellas se aplica de acuerdo a las características del software a desarrollar.¹

2.1.1 ANÁLISIS COMPARATIVO DE METODOLOGÍAS DE PROGRAMACIÓN.

Una vez analizadas las metodologías, se resume sus cualidades en la Tabla 2-1.

Características	RUP ^(*)	SCRUM ^(*)	XP ^(*)
Retroalimentación en realización de pruebas de software	8	6	9
Comunicación y trabajo en equipo entre cada fase de la metodología	8	8	10
Desarrollo efectivo de pruebas unitarias y de integración.	7	5	9
Trabajo en conjunto entre el cliente activo y los desarrolladores del sistema.	5	4	10
Agilidad y flexibilidad en el desarrollo funcional y operacional del sistema.	7	6	9
Independiente de las tecnologías.	5	7	10
Manejo eficaz del tiempo de ejecución entre fases.	6	5	9
Respuesta positiva a los	7	8	10

¹Instituto Tecnológico Superior de Apatzingán, Metodologías de desarrollo de software, Alejandra Virrueta Méndez, Diciembre, 2010.

cambios entre fases de la metodología.			
---	--	--	--

(*) Valor del puntaje en el análisis: Sobre 10 puntos.

Tabla 2-1: Metodologías de Desarrollo de Software.

Fuente: El Autor, 2013

2.1.2 RESULTADO DE METODOLOGÍA.

Según el análisis obtenido se puede interpretar que la metodología XP se enfoca en la necesidad del cliente dando una disposición completa a la hora de cambios, sabemos que hoy en día el tiempo es un recurso muy importante y los clientes necesitan productos de calidad y no documentación obsoleta es por ello que se escogió esta metodología ya que se enfoca en el proyecto a ser entregado más que en la documentación redactada, además de los cambios o necesidades que se pueden dar en el momento del desarrollo.

A continuación se detallará de una manera específica las fases así como ciclo de vida y desarrollo de la metodología XP.

2.1.3 METODOLOGÍA DE DESARROLLO XP

Actualmente XP es una de las metodologías de mayor aceptación en la industria del software, su enfoque basado en los métodos ágiles, su énfasis en la gestión del recurso humano el cuál es uno de los puntos más críticos en todo proyecto, y sus principios de previsibilidad y adaptabilidad hacen de esta metodología una buena opción a seguir.²XP fue introducida como metodología ágil de desarrollo de software sobre finales de los 90s. Uno de los conocidos “caso de éxito” fue publicado a fines de 1998, cuando Kent Beck introdujo la nueva metodología en el proyecto de desarrollo.

“Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar.”[3]

²Cuatro enfoques metodológicos para el desarrollo de Software RUP-MSF-XP-SCRUM, Oliver Andrés Pérez A., Junio 10, 2011.

2.1.4 FASES DE LA METODOLOGÍA XP

A continuación se detalla cada una de las fases que se relacionan de forma iterativa como lo muestra la Figura 2-1.



Figura 2-1: Extreme Programming.
Fuente: Introducción a Extreme Programming.
Autor: Fernández Gerardo.

2.1.5 PLANIFICACIÓN

La metodología XP plantea la planificación como un dialogo continuo entre las partes involucradas en el proyecto, incluyendo al cliente, a los programadores y a los coordinadores o gerentes. El proyecto comienza recopilando “Historias de usuarios”, una vez obtenidas las “historias de usuarios”, se realizan pequeños programas de prueba (“spikes”), para reducir estos riesgos. Una vez realizadas estas estimaciones, se organiza una reunión de planificación, con los diversos actores del proyecto (cliente, desarrolladores, gerentes), a los efectos de establecer un plan o cronograma de entregas (Release Plan) en los que todos estén de acuerdo. Una vez acordado este cronograma, comienza una fase de iteraciones, en dónde en cada una de ellas se desarrolla, prueba e instala unas pocas “historias de usuarios”.

Los conceptos básicos de esta planificación son los siguientes:

Historias de usuarios

Las Historias de usuarios sustituyen a los documentos de especificación funcional, y a los casos de uso. Estas historias son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. Las historias de usuario deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando llegue el momento de la implementación, los desarrolladores dialogarán directamente con el cliente para obtener todos los detalles necesarios.

Las historias de usuarios deben poder ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra historia.

Plan de entregas (Release Plan)

El cronograma de entregas establece qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas. Este cronograma será el resultado de una reunión entre todos los actores del proyecto (cliente, desarrolladores, gerentes, etc.). XP denomina a esta reunión Juego de planeamiento (Planning game), pero puede denominarse de la manera que sea más apropiada al tipo de empresa y cliente (por ejemplo, Reunión de planeamiento, Planning meeting o Planning workshop)

Típicamente el cliente ordenará y agrupará según sus prioridades las historias de usuario. El cronograma de entregas se realiza en base a las estimaciones de tiempos de desarrollo realizadas por los desarrolladores.

Luego de algunas iteraciones es recomendable realizar nuevamente una reunión con los actores del proyecto, para evaluar nuevamente el plan de entregas y ajustarlo si es necesario.

Plan de iteraciones (Iteration Plan)

Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido. Al comienzo

de cada ciclo, se realiza una reunión de planificación de la iteración. Cada historia de usuario se traduce en tareas específicas de programación.

Asimismo, para cada historia de usuario se establecen las pruebas de aceptación. Las pruebas de aceptación que hayan fallado en el ciclo anterior son analizadas para evaluar su corrección, así como para prever que no vuelvan a ocurrir.

Reuniones diarias de seguimiento (Stand-up meeting)

El objetivo de tener reuniones diarias es mantener la comunicación entre el equipo, y compartir problemas y soluciones. En la mayoría de estas reuniones, gran parte de los participantes simplemente escuchan, sin tener mucho que aportar. Para no quitar tiempo innecesario del equipo, se sugiere realizar estas reuniones en círculo y de pie.

2.1.6 DISEÑO.

La metodología XP hace especial énfasis en los diseños simples y claros. Los conceptos más importantes de diseño en esta metodología son los siguientes:

Simplicidad

Un diseño simple se implementa más rápidamente que uno complejo. Por ello XP propone implementar el diseño más simple posible que funcione.

Soluciones (*spike*)

Cuando aparecen problemas técnicos, o cuando es difícil de estimar el tiempo para implementar una historia de usuario, pueden utilizarse pequeños programas de prueba (llamados *spike*), para explorar diferentes soluciones. Estos programas son únicamente para probar o evaluar una solución, y suelen ser desechados luego de su evaluación.

Recodificación

La recodificación (refactoring) consiste en escribir nuevamente parte del código de un programa, sin cambiar su funcionalidad, a los efectos de hacerlo más simple, conciso y/o entendible. Muchas veces, al terminar de escribir un código de programa, pensamos que, si lo comenzáramos de nuevo, lo hubiéramos hecho en

forma diferente, más clara y eficientemente. Sin embargo, como ya está pronto y funciona, rara vez es reescrito. Las metodologías de XP sugieren recodificar cada vez que sea necesario. Si bien, puede parecer una pérdida de tiempo innecesaria en el plazo inmediato, los resultados de ésta práctica tienen sus frutos en las siguientes iteraciones, cuando sea necesario ampliar o cambiar la funcionalidad. La filosofía que se persigue es, como ya se mencionó, tratar de mantener el código más simple posible que implemente la funcionalidad deseada.

Metáforas

Una metáfora es algo que todos entienden, desarrollada por los programadores al inicio del proyecto, define una historia de cómo funciona el sistema completo. La metáfora expresa la visión evolutiva del proyecto que define el alcance y propósito del sistema. Las tarjetas **CRC** también ayudarán al equipo a definir actividades durante el diseño del sistema. Cada tarjeta representa una clase en la programación orientada a objetos y define sus responsabilidades (lo que ha de hacer) y las colaboraciones con las otras clases (cómo se comunica con ellas). Para poder diseñar el sistema como un equipo, se deben cumplir tres requisitos: Clase, Responsabilidad y Colaboración (CRC), estas tarjetas CRC permitirán desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos, permiten también que el equipo completo contribuya en la tarea del diseño.

Las principales características de las tarjetas son:

- Identificación de clases y asociaciones que participan del diseño del sistema.
- Obtención de las responsabilidades que debe cumplir cada clase.
- Establecimiento de cómo una clase colabora con otras clases para cumplir con sus responsabilidades.

2.1.7 DESARROLLO.

Disponibilidad del cliente

Uno de los requerimientos de XP es tener al cliente disponible durante todo el proyecto. No solamente como apoyo a los desarrolladores, sino formando parte

del grupo. El involucramiento del cliente es fundamental para que pueda desarrollarse un proyecto con la metodología XP.

Al comienzo del proyecto, el cliente debe proporcionar las historias de usuarios.

No se requieren de largos documentos de especificaciones, sino que los detalles son proporcionados por el cliente, en el momento adecuado, “cara a cara” a los desarrolladores.

Programación dirigida por las pruebas (Test-driven programming)

En las metodologías tradicionales, la fase de pruebas, incluyendo la definición de los test, es usualmente realizada sobre el final del proyecto, o sobre el final del desarrollo de cada módulo. La metodología XP propone un modelo inverso, en el que, lo primero que se escribe son los test que el sistema debe pasar. Luego, el desarrollo debe ser el mínimo necesario para pasar las pruebas previamente definidas.

Las pruebas a los que se refieren esta práctica, son las pruebas unitarias, realizados por los desarrolladores. La definición de estos test al comienzo, condiciona o “dirige” el desarrollo.

Programación en pares

XP propone que se desarrolle en pares de programadores, ambos trabajando juntos en un mismo ordenador. Si bien parece que ésta práctica duplica el tiempo asignado al proyecto (y por ende, los costos en recursos humanos), al trabajar en pares se minimizan los errores y se logran mejores diseños, compensando la inversión en horas. El producto obtenido es por lo general de mejor calidad que cuando el desarrollo se realiza por programadores individuales.

Integraciones permanentes

Todos los desarrolladores necesitan trabajar siempre con la última versión.

XP promueve publicar lo antes posible las nuevas versiones, aunque no sean las últimas, siempre que estén libres de errores. Idealmente, todos los días deben existir nuevas versiones publicadas.

Para evitar errores, solo una pareja de desarrolladores puede integrar su código a la vez.

Ritmo sostenido

La metodología XP indica que debe llevarse un ritmo sostenido de trabajo.

Anteriormente, ésta práctica se denominaba *Semana de 40 horas*. Sin embargo, lo importante no es si se trabajan, 35, 40 o 42 horas por semana. El concepto que se desea establecer con esta práctica es el de planificar el trabajo de manera que se lleve un ritmo constante y razonable, sin sobrecargar al equipo.

Cuando un proyecto se retrasa, trabajar tiempo extra puede ser más perjudicial que beneficioso. El trabajo extra desmotiva inmediatamente al grupo e impacta en la calidad del producto. En la medida de lo posible, se debería renegociar el plan de entregas (Release Plan), realizando una nueva reunión de planificación con el cliente, los desarrolladores y los gerentes. Adicionalmente, agregar más desarrolladores en proyectos ya avanzados no siempre resuelve el problema.

2.1.8 PRUEBAS

Pruebas unitarias

Las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Por otra parte, como se mencionó anteriormente, las pruebas deben ser definidas antes de realizar el código (*Test-driven programming*).

Detección y corrección de errores

Cuando se encuentra un error (*bug*), éste debe ser corregido inmediatamente, y se deben tener precauciones para que errores similares no vuelvan a ocurrir.

Asimismo, se generan nuevas pruebas para verificar que el error haya sido resuelto.

Pruebas de aceptación

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada.

Las pruebas de aceptación son consideradas como *pruebas de caja negra* (Black box system tests). Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución.

Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación.³⁴

2.2 MODELO DE PROCESO DE SOFTWARE.

Sommerville⁵ define modelo de proceso de software como *“Una representación simplificada de un proceso de software, representada desde una perspectiva específica. Por su naturaleza los modelos son simplificados, por lo tanto un modelo de procesos del software es una abstracción de un proceso real”*.

2.2.1 UML.

El lenguaje unificado de diagrama o notación (UML) sirve para especificar, visualizar y documentar esquemas de sistemas de software orientado a objetos. UML no es un método de desarrollo, lo que significa que no sirve para determinar qué hacer en primer lugar o cómo diseñar el sistema, sino que simplemente le ayuda a visualizar el diseño y a hacerlo más accesible para otros.

UML está diseñado para su uso con software orientado a objetos, y se compone de muchos elementos de esquematización que representan las diferentes partes de un sistema de software. Los elementos UML se utilizan para crear diagramas, que representa alguna parte o punto de vista del sistema.

³ Jose Joskowicz, *Reglas y Prácticas en eXtreme Programming.*, 10/02/2008.

⁴ Kent Beck, *Una explicación de la Programación extrema:* www.xprogramming.com.

⁵ SOMMERVILLE, I, *Ingeniería de Software*, Person Educación, 2002.

2.2.2 MODELAMIENTO DE CLASES.

Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenido.

Un diagrama de clases está compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

Elementos

Clase.

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

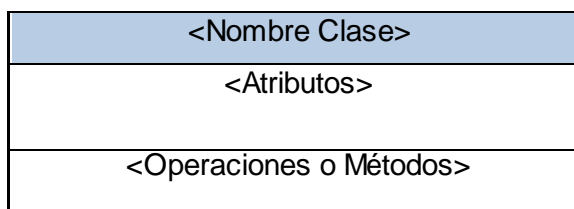


Figura 2-2: Representación Gráfica de una Clase en UML
Fuente: El Autor, 2013

En donde:

- **Superior:** Contiene el nombre de la Clase
- **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser *private*, *protected* o *public*).
- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: *private*, *protected* o *public*).

Atributos.- Características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno.

Métodos.- Operaciones de una clase son la forma en como ésta interactúa con su entorno.

Visibilidad.- Los atributos y los métodos tienen las siguientes características.

- **public (+):** Indica que será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.
- **private (-):** Indica que sólo será accesible desde dentro de la clase (sólo sus métodos lo pueden acceder).
- **protected (#):** Indica que no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de las subclases que se deriven.

Relaciones entre Clases:

Ahora ya definido el concepto de Clase, es necesario explicar cómo se pueden interrelacionar dos o más clases (cada uno con características y objetivos diferentes).

Antes es necesario explicar el concepto de cardinalidad de relaciones: En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

- **uno o muchos:** 1..* (1..n)
- **0 o muchos:** 0..* (0..n)
- **número fijo:** m (m denota el número).

Herencia (Especialización/Generalización).- Indica que una subclase hereda los métodos y atributos especificados por una Súper Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Súper Clase (public y protected).

Agregación.- es un tipo especial de asociación donde se añade el matiz semántico de que la clase de donde parte la relación representa el “todo” y las clases relacionadas “las partes”.

Realmente Java y la mayoría de lenguajes orientados a objetos no disponen de una implementación especial para este tipo de relaciones. Básicamente se tratan como las asociaciones ordinarias.

Asociación.- La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre sí. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.

Dependencia o Instanciación (uso).- Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase). Se denota por una flecha punteada.

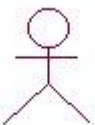
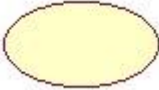
El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra.⁶

2.2.3 CASOS DE USO

El diagrama de casos de uso representa la forma en como un Cliente (Actor) opera con el sistema en desarrollo, además de la forma, tipo y orden en como los elementos interactúan (operaciones o casos de uso).⁷

Un diagrama de casos de uso consta de los siguientes elementos:

- Actor.
- Casos de Uso.
- Relaciones de Uso, Herencia y Comunicación.

Elementos	Utilidad	Denotación
Actor	Es un rol que un usuario desempeña en el sistema	
Caso de Uso	Es una tarea u operación en la cual se especifica lo que se realizara tras dicha orden puede	

⁶Patricio Salinas Caro. www.dcc.uchile.cl. [Online]. <http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>.

⁷Patricio Salinas Caro. www.dcc.uchile.cl. [Online]. <http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>



		ser enviada desde un agente externo, un actor o también desde otro caso de uso.		
Relaciones	Asociación	Una de las relaciones más básica, indica la llamada desde un actor o caso de uso a otro caso de uso		
	Dependencia o Instanciación	Se aplica cuando una clase depende de otra		
	Generalización	Uno de los más usados cumple una doble función, las cuales son: uses y extends	Uses Extends	Útil cuando se tiene { } de características similares en un caso de uso Útil cuando un caso de usos es similar es similar a otro con respecto a sus características

Tabla 2-2: Elementos Necesarios para un Diagrama de Casos de Uso.
Fuente: El Autor, 2013

2.2.4 DIAGRAMA DE INTERACCIÓN.

El diagrama de interacción, representa la forma en cómo un Cliente (Actor) u Objetos (Clases) se comunican entre sí en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.

Dicho diagrama puede ser obtenido de dos partes, desde el Diagrama Estático de Clases o el de Casos de Uso (son diferentes). [4]

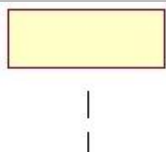
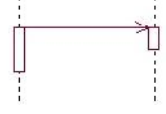
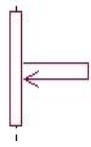
Elementos	Utilidad	Denotación
Un objeto o actor	El rectángulo representa una instancia de un objeto en particular, y la línea punteada representa las llamadas a métodos del objeto	
Mensaje de un objeto a otro objeto	Se representa por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular	
Mensaje de un objeto a si mismo	No solo llamadas a métodos de objetos externos pueden realizarse, también es posible visualizar llamadas a métodos desde el mismo objeto en estudio.	

Tabla 2-3: Componentes de un Diagrama de Interacción.

Fuente: El Autor, 2013

2.3 SOFTWARE LIBRE

Software Libre es una forma ética de entender el uso libre que le da el usuario a cualquier programa de software como procesador de texto o de cálculo, que puede ser utilizado, modificado, mejorado, copiado y distribuido de manera libre de acuerdo con los valores de: colaboración, competitividad, eficiencia, privacidad, no discriminación y sobre todo libertad a que nadie puede restringir el desarrollo, comercialización y distribución. Para así promover la cooperación en la sociedad. La inclinación a utilizar software libre se basa en muchas características ventajas y desventajas del software propietario, continuación podremos visualizar un cuadro que ayude a entender por qué utilizar software libre y no propietario.

Características	Software Libre	Software Propietario
Costo de adquisición	No	Si
Uso de centros de inclusión digital gratuitos	Si	No
Cambios en códigos de programación	Si	No
Dependencia de proveedor.	No	Si

Soporte y actualización gratuita y a bajo costo	Si	No
Monopolización del mercado informático	No	Si
Aprendizaje en técnicas de programación	Si	No
Requisitos costosos de hardware	No	Si
Costo de adquisición	No	Si

Tabla 2-4: Comparación de Software Propietario y Libre
Fuente: El Autor, 2013

Como podemos observar el software libre da la libertad de jugar con código y poderlo mejorar dando así mayor libertad a la hora de programar y realizar los requerimientos necesarios para mi proyecto de titulación. Una de las ventajas más importantes de utilizar software libre es que el costo de licencias se reduce a 0.

2.3.1 LIBERTADES DEL SOFTWARE LIBRE

Los programas que usan software libre se basan en las siguientes libertades que son:

- **Libertad de uso:** el usuario tiene libre albedrio para ejecutar su programa de software de acuerdo a sus necesidades.
- **Libertad de estudio:** el usuario puede conocer, comprender como funciona el programa y al mismo tiempo poder adaptarle como se lo requiera.
- **Libertad de modificación:** el usuario puede hacerle mejoras al programa y puede compartir dichas mejoras para que otros usuarios puedan también mejorarlo.
- **Libertad de distribución:** el usuario puede distribuir su programa por medio de la copia, venta o distribución gratuita.⁸

2.3.2 TIPOS DE LICENCIAS

Las licencias son contratos o acuerdos entre el creador o autor del programa que establece cómo el usuario puede usar, modificar, copiar ese software con las empresas que están interesadas en la utilización del software.

⁸http://es.wikipedia.org/wiki/Software_libre

Se puede distinguir diferentes tipos de licencias de software libre ya que existen licencias que son limitadas y tienen restricciones. Algunas de éstas son:

Licencia GNU GPL

Las siglas de esta licencia significan: **Licencia Pública General de GNU**. Le permite al usuario usar, distribuir y modificar el software siempre y cuando la siguiente versión que se haya generado de ese software se haga dentro de la misma licencia. Esto conllevaría a que esta licencia no pueda sea compatible si se deseara trabajar en conjunto con un software privativo o que pertenezca a otro tipo de licencia.

Cuando hablamos de software libre, estamos refiriéndose a libertad, no a precio. Nuestras Licencias Públicas Generales están diseñadas para asegurar de que se tenga la libertad de distribuir copias de software libre (y cobrar por ese servicio si quiere), de que reciba el código fuente o que pueda conseguirlo si lo quiere, de que pueda modificar el software o usar fragmentos de él en nuevos programas libres, y de que sepa que puede hacer todas estas cosas.

2.3.3 VENTAJAS

El usuario posee la libertad de darle cualquier tipo de uso al software libre por esta licencia y no tiene la obligación de proteger el resultado dependiendo de un proveedor. Los programas de software libre se encuentran en constante evolución ya que los programadores mejoran el código mediante la compartición de posibles soluciones a los errores de los programas y colaborando con otros colegas.

La seguridad al momento de guardar información muy valiosa u obtenerla mediante conexión a internet es muy eficaz. El uso de antivirus de software gratuito es una solución muy económica ya que no se tendrá que pagar la cuota anual o mensual que se suele cobrar por la compra del software de un almacén externo. Se puede sustituir por otro, siempre que lo desee, ya que todos son libres. Así, usted puede proteger su equipo de forma gratuita.

2.3.4 DESVENTAJAS.

La principal desventaja de no usar software propietario es en el momento de pedir soluciones al instante y no tener que preocuparse por ser un experto en cada tema, esto se lo podría conseguir con un soporte al instante y en cualquier parte del mundo para que la herramienta no quede a la espera de que alguien brinde una solución.

Por otra parte, es más difícil el entendimiento del software pues no es muy intuitivo, no ofrece muchas garantías y falta de compatibilidad de hardware.

2.3.5 SOFTWARE LIBRE EN ECUADOR

En Ecuador se ha venido promoviendo de una manera muy exitosa del uso de software libre, tanto es así que el día jueves 10 de Abril del 2008 el presidente de la República del Ecuador Eco. Rafael Correa Delgado emitió el decreto 1014 que promueve el uso de software libre en las instituciones públicas del Ecuador.

Desde esta fecha hasta la actualidad la mayoría de las instituciones públicas empezaron a migrar sus aplicaciones, bases de datos a software libre y poco a poco los funcionarios públicos se adaptaron mediante capacitaciones y además se obtuvieron grandes resultados en el proceso de liberación de las instituciones del Ecuador en el software propietario es la ratificación formal de OpenDocument 1.0 por el INEN (Instituto Ecuatoriano de Normalización.)

El Sistema de Contratación pública fue construido con herramientas de Software Libre y desarrollado por profesionales cien por ciento ecuatorianos, demostrando las posibilidades y capacidades tecnológicas comprometidas con el progreso del país.

De esta manera se cumplen los lineamientos del Gobierno ecuatoriano contenidos en el decreto 1014, que *“establece como política de la Administración Central el uso de Software Libre en todas sus instancias”*.

Mediante el Software Libre se puede acceder a los códigos fuente y mejorar sus aplicaciones. En cuanto a seguridad informática, tener libre acceso a programas y datos.

En el aspecto económico la migración d software privativo a software libre ha hecho que el Ecuador obtenga ahorros significativos de alrededor de 15 millones en el costo de licencias y la generación de empleo para los profesionales ecuatorianos, así como también libera de la dependencia de herramientas informáticas producidas por las grandes transnacionales.

“Es importante que seamos productores de tecnología, no simples consumidores; debemos tender a ser dueños de los códigos fuente, para desarrollar muchos productos informáticos que sin duda serán de suma utilidad para las empresas públicas y privadas en América Latina”. [5]

2.4 SISTEMAS OPERATIVOS PARA SERVIDORES

Un servidor es un equipo informático que forma parte de una red y provee servicios a otros equipos cliente.

Se denomina servidor dedicado, aquel que dedica todos sus recursos a atender solicitudes de los equipos cliente, y además de ser robusto requiere estar activo 24 horas 7 días a la semana lo que se denomina 24/7, con un tiempo mínimo de mantenimiento.

Existen gran cantidad de tipos de servidores o roles que estos pueden desempeñar, entre los cuales vamos a explicar los servicios que necesitamos implementar en nuestro servidor.

- Servidor Web: Almacena contenido Web y lo pone al servicio de aquellos usuarios que lo solicitan.
- Servidor de Base de Datos: es aquel que provee servicios de base de datos a otros programas o equipos cliente.
- Servidor FTP: su función es permitir el intercambio de ficheros entre equipos, normalmente su aplicación va muy ligada a los servidores Web.

Según los requerimientos de servicios necesitados vamos a citar sistemas operativos que se adecuen al proyecto de titulación en el que se está trabajando.

2.4.1 CENTOS

CentOS es un sistema operativo que opera como un servidor de distribución. Se basa en Red Hat Enterprise Linux (RHEL), que usa algunos paquetes del sistema operativo Fedora; una versión beta pública de Red Hat Enterprise Linux.

La utilización de CentOS es la mejor opción en sistemas operativos para usar en los servidores debido a la orientación de sus distribuciones. Cualquier problema de soporte no está cubierto por CentOS, será cubierto por RHEL.

Este sistema cuenta con un excelente soporte y estabilidad siendo así muy necesaria para las grandes empresas que manejan una gran carga de datos y programas.

El usuario tendrá la certeza de que el sistema es estable, rápido y casi completamente libre de errores.

Las versiones nuevas de CentOS son liberadas cada 2 años y actualizadas regularmente (cada 6 meses) para el soporte de hardware nuevo.[6]

2.4.2 ANÁLISIS COMPARATIVO ENTRE SERVIDORES.

En la siguiente tabla se muestra las diferencias y requerimientos de instalación entre los servidores que se ha mencionado anteriormente, mediante un análisis se ha tenido en cuenta los sistemas operativos que se adecuan a los equipos con que se cuentan.

	Windows Server 2003	Windows Server 2008	Ubuntu	CentOS
Modelo de desarrollo	Software Propietario Shared Source	Software propietario; Shared Source	Software libre y de código abierto	Software libre
Última versión estable	5.2 ServicePack 2 en marzo del 2007	6.0 ServicePack 2 (SP2) en abril del 2009	12.10 (Quantal Quetzal) en octubre del 2012	6.3 en julio del 2012

Núcleo		NT 5.2	NT 6.1	Linux	Linux
Tipo de Núcleo		Hibrido	Hibrido	Monolítico	Monolítico
Requisitos de Hardware	Procesador:	500 Mhz	1 GHz(x86) o 1.4 GHz (x64)	700 Mhz	i386, x86_64
	Memoria	256 MB	512 MB	512 MB	64 MB
	RAM:	4 Gb de	10 GB de	5 GB de	1 GB
	Disco Duro:	espacio libre	espacio libre	espacio libre	

Tabla 2-5 Requerimientos de Instalación del Servidor

Fuente: El Autor, 2013

Un Sistema Operativo es el software encargado de ejercer el control y coordinar el uso del hardware entre diferentes programas de aplicación y los diferentes usuarios. Es un administrador de los recursos de hardware del sistema.

A continuación en una tabla se explica las características que se debe necesitar para comparar entre los servidores que se acoplan más a las necesidades de la aplicación a usar, con un puntaje dado de acuerdo a un estudio realizado de acuerdo a valores altos (10) y bajos (1) con los diferentes servidores que más utilizan las empresas de desarrollo de software.

ASPECTOS	WINDOWS (°)	CENTOS(°)	UBUNTU(°)
Ataques por Virus	9	1	1
Costo por distribución	9	1	1
Costo de Mantenimiento	5	5	5
Requerimientos de hardware	10	5	8
Documentación	10	8	8
Facilidad de Uso	8	5	8
Entorno Amigable	8	8	8
Rapidez	7	9	8
Fiabilidad, Disponibilidad, y Escalabilidad	8	10	9
Compatibilidad con Hardware	9	7	5

Rendimiento de Inicio y Apagado	6	8	7
Rendimiento del Sistema Operativo	5	9	7
Gestor de Usuarios	5	8	8
Aprovechamiento de RAM	4	8	7
Uso de Disco Duro	8	4	5

(*) Valor del puntaje en el análisis: Sobre 10 puntos.

Tabla 2-6: Análisis de Sistemas Operativos

Fuente: El Autor, 2013.

2.4.3 RESULTADO DEL ANALISIS

Con la tabla comparativa se puede observar que el sistema operativo CentOS, tiene mayores ventajas y al ser de licencia libre se ajusta a nuestras posibilidades y requerimientos de hardware, a más de un uso muy alto a nivel mundial como software diseñado para trabajar en servidores, por la experiencia y el soporte que puede brindar este se ha decidido tomar como solución el uso de CentOS como sistema operativo, teniendo en cuenta que la aplicación a desarrollar será multiplataforma la cual funcionará sin ningún inconveniente en cualquiera de los servidores antes mencionados.

2.5 LENGUAJE DE PROGRAMACIÓN

Los lenguajes de programación fueron diseñados para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo.

Características	JAVA	PHP	C#
Multiplataforma	9	9	3
Seguridad	8	8	8
Orientado a Objetos	9	9	7
Soporte con bdd	9	8	9

Código Abierto	9	9	3
----------------	---	---	---

Tabla 2-7: Lenguajes de Programación más Utilizados

Fuente: El Autor, 2013

2.5.1 JAVA

Java es un lenguaje de programación que difiere de la mayoría de los otros lenguajes de programación ya que los programas en Java se pueden ejecutar en diferentes tipos de ordenadores, es decir es multiplataforma y esto fue una ventaja crucial en el momento de orientarnos al desarrollo de la aplicación, tomando en cuenta que en la actualidad la mayoría de las aplicaciones en el mundo real tienen una gran demanda por software desarrollado en java, de esta manera se especificarán algunos de los complementos con los que se debe contar para una que aplicación pueda funcionar correctamente.

La versión actual hace de Java uno de los lenguajes de programación más populares y apoyado en los ordenadores personales, teléfonos celulares, PDAs y otros dispositivos pequeños. Recientemente, Java fue liberado bajo la licencia de código abierto GPL.

2.5.2 JSF (JAVA SERVER FACES)

JSF es un nuevo estándar de framework, que hace que sea fácil de construir interfaces de usuario para las aplicaciones Java Web mediante el modelamiento de componentes reutilizables en una página, basándose en el Modelo-Vista-Controlador, patrón de diseño (MVC).

JSF es como una caja de herramientas que está lleno de componentes listos para usar, donde se puede añadir rápidamente y reutilizar estos componentes en una página Web capturando eventos generados por las acciones de estos componentes.

Por lo general, se integran los componentes en una página JSP con etiquetas personalizadas definidas por la tecnología JSF y utilizar el framework para manejar la navegación de una página a otra.

Los componentes se pueden anidar dentro de otro componente, por ejemplo, dentro de un formulario puede encontrarse una caja de texto y un botón.

Para mostrar un botón con un evento en el JSF es bastante complejo en comparación con un ejemplo puro HTML que muestra un botón.

Beneficios de JSF

Facilita el desarrollo de aplicaciones Web basado en tecnología Java. Éstos son algunos de los beneficios del uso de JSF:

- Proporciona componentes estándar reutilizables para la creación de interfaces de usuario para aplicaciones Web.
- Cuenta con muchas bibliotecas de etiquetas para acceder y manipular los componentes.
- Guarda automáticamente los datos del formulario y rellena el formulario cuando se muestra en el lado del cliente.
- Encapsula el manejo de eventos y la lógica de representación de componentes de los programadores, los programadores sólo tiene que utilizar los componentes personalizados.
- JSF es un estándar y los proveedores pueden desarrollar las implementaciones de JSF.

Componentes JSF

JSF incluye principalmente:

- Conjunto de APIs para representar y administrar el estado de los componentes que ayuda a la validación del lado del servidor, gestión de eventos, la página de navegación, la conversión de datos, etc. Biblioteca de etiquetas JSP personalizada para crear componentes de interfaz de usuario en una página de vista.
- La interfaz de usuario (interfaz de usuario) creados con tecnología JSF ejecuta en el servidor y el resultado se muestra en el cliente. Objetivo de JSF es crear aplicaciones Web más rápidas y más fáciles.

2.5.3 PERSISTENCIAS

Cuando abordamos el desarrollo de una aplicación en Java, uno de los primeros requerimientos que se debe resolver es la integración con una base de datos para guardar, actualizar y recuperar la información que utiliza nuestra aplicación.

Se llama “persistencia” de los objetos a su capacidad para guardarse y recuperarse desde un medio de almacenamiento. La persistencia en Base de Datos relacionales se suele implementar mediante el desarrollo de funcionalidad específica utilizando la tecnología JDBC o mediante frameworks que automatizan el proceso a partir de mapeos (conocidos como Object Relational Mapping, ORM) como es el caso de Hibernate.⁹

Es la capacidad del programador para conseguir que sus datos sobrevivan a la ejecución del proceso que los creo, de forma que puedan ser reutilizados en otro proceso. Cada objeto, independiente de su tipo, debería poder llegar a ser persistente sin traducción explícita. También, debería ser implícito que el usuario no tuviera que mover o copiar los datos expresamente para ser persistentes.[7]

Esta definición recuerda qué es tarea del programador, determinar cuándo y cómo una instancia pasa a ser persistente o deja de serlo, o cuando, debe ser nuevamente reconstruida; asimismo, que la transformación de un objeto en su imagen persistente y viceversa, debe ser transparente para el programador, sin su intervención; y que todos los tipos, clases, deberían tener la posibilidad de que sus instancias perduren.

Así, en vista de lo anterior, la persistencia de los objetos es, como su nombre indica, el almacenamiento de los objetos.

Este hecho permite que siempre se mantengan disponibles para la aplicación, para que esta los pueda consultar, modificar, combinar, etc. Es decir, gestionarlos. La persistencia de los objetos debe conservar dicho objeto en el estado que el usuario considere interesante para sus objetivos.

⁹<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=PersistenciaJava>

2.5.3.1 EVALUACIÓN DE TECNOLOGÍAS EXISTENTES.

Características	HIBERNATE ^(*)	JPA ^(*)	JDBC ^(*)
Software Propietario	3	8	7
Recuperación de Datos	9	8	5
Flexibilidad	9	7	6
Simple	8	9	7
Múltiples motores Base datos	8	5	9
Volúmenes alto de Información	9	9	7
Documentación	8	8	9
Mapeo objeto/relacional	9	7	3

(*) Valor del puntaje en el análisis: Sobre 10 puntos.

Tabla 2-8: Características de Tecnologías Persistentes.

Fuente: El Autor, 2013

2.5.3.2 HIBERNATE

Hibernate es un framework que no sólo se ocupa del mapeo desde las clases Java a las tablas de las bases de datos (y desde los tipos de datos de Java a los tipos de datos de SQL), sino que también facilita la consulta y recuperación de datos. Esto puede reducir de manera importante el tiempo de desarrollo que se tomaría con el manejo de datos de forma manual en SQL y JDBC. Hibernate busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información Hibernate le permite a la aplicación manipular los datos de la base operando sobre objetos, con todas las características de la POO¹⁰. Hibernate convertirá los datos entre los tipos utilizados por Java y los definidos por SQL. Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución.

Hibernate ofrece también un lenguaje de consulta de datos llamado HQL (Hibernate Query Language), al mismo tiempo que una API para construir las consultas programáticamente (conocida como "criteria").

¹⁰ POO (Programación Orientada a Objetos).

Hibernate para Java puede ser utilizado en aplicaciones Java independientes o en aplicaciones Java EE, mediante el componente Hibernate Annotations que implementa el estándar JPA, que es parte de esta plataforma.[8]

Se basa en:

- JavaBeans: que son clases para persistir.
- Archivo XML: por cada una de sus clases que indica el mapping entre objeto y relaciones (MiClase.hbm.xml).
- Hibernate.cfg.xml: se crea y configura este archivo del framework hibernate.

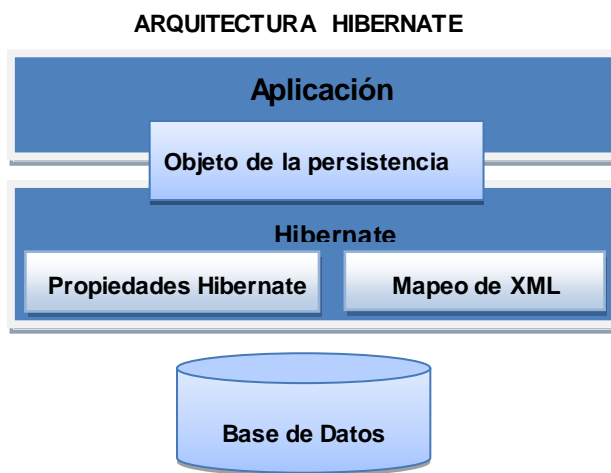


Figura 2-3: Arquitectura Hibernate.

Fuente: El Autor, 2013

JavaBean: Un POJO es similar a una Java Bean, con propiedades accesibles mediante métodos setter y getter, como por ejemplo: TbUsuario.java

```

package pojos;
public class TbUsuario implements java.io.Serializable {

    private String idUsuario;
    private TbPerfil tbPerfil = new TbPerfil();
    private String apellidos;
    private String nombres;

    //Setters y Gettes
}
  
```

Figura 2-4: Ejemplo Java Bean.

Fuente: El Autor, 2013

Hibernate Query Language(HQL): es un lenguaje de consulta, que permite hacer preguntas basadas en los objetos y sus propiedades.

```
Ejemplos con SQL:  
From Empleado e where e.id = 1  
En SQL podría ser:  
Select id, nombre,apellidos from empleados where id = 1
```

Figura 2-5: Ejemplo HQL.

Fuente: El Autor, 2013

2.5.4 SERVIDOR DE APLICACIONES

El concepto de servidor de aplicaciones está relacionado con el concepto de sistema distribuido. Un sistema distribuido, permite mejorar tres aspectos fundamentales en una aplicación: la alta disponibilidad, la escalabilidad y el mantenimiento.

- **Alta disponibilidad** hace referencia a que un sistema debe estar funcionando las 24 horas del día los 365 días al año.
- **Escalabilidad** es la capacidad de hacer crecer un sistema cuando se incrementa la carga de trabajo (el número de peticiones). Cada máquina tiene una capacidad finita de recursos y por lo tanto sólo puede servir un número limitado de peticiones. Si, por ejemplo, tenemos una tienda que incrementa la demanda de servicio, debemos ser capaces de incorporar nuevas máquinas para dar servicio.
- **Mantenimiento** tiene que ver con la versatilidad a la hora de actualizar, depurar fallos y mantener un sistema. La solución al mantenimiento es la construcción de la lógica de negocio en unidades reusables y modulares.

El estándar J2EE¹¹ permite el desarrollo de aplicaciones de empresa de una manera sencilla y eficiente. Una aplicación desarrollada con las tecnologías J2EE permite ser desplegada en cualquier servidor de aplicaciones o servidor Web que cumpla con el estándar. Un servidor de aplicaciones es una implementación de la especificación J2EE. La arquitectura J2EE es la siguiente:

¹¹ J2EE (Java 2 PlatForm Enterprise Edition su traducción es Java Empresarial).

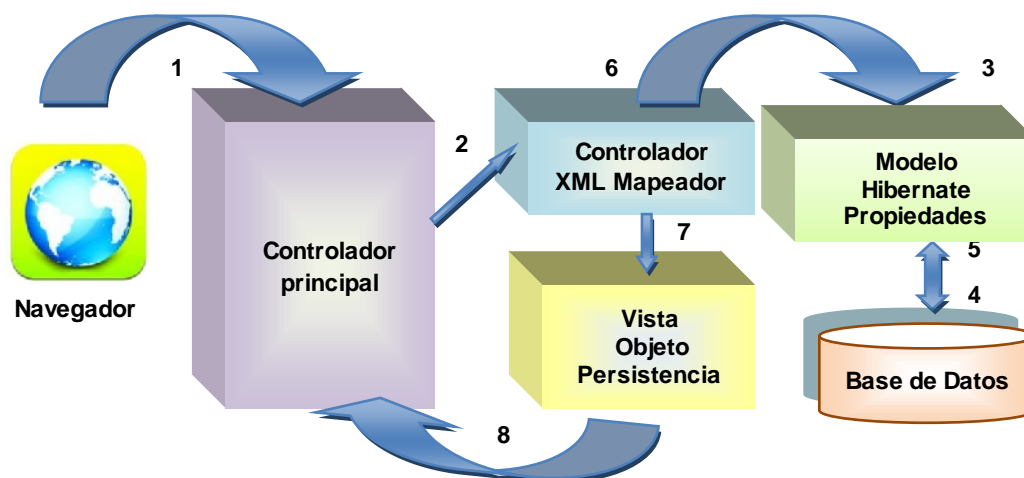


Figura 2-7: Arquitectura MVC
Fuente: El Autor, 2013

El uso de un servidor de aplicaciones como GlassFish es muy común para aplicaciones y en especial para JSF, ya que funciona de manera conjunta con persistencias, una buena consola de administración, código abierto e integración con Netbeans, estas son algunas de las razones necesarias que se tienen para inclinarse por este servidor de aplicaciones, de la misma manera cuenta con un muy alto soporte, documentación y seguridad.[9]

2.5.4.1 GLASSFISH

Es un servidor de aplicaciones que implementa la plataforma JavaEE5, por lo que soporta las últimas versiones de tecnologías como: JSP, JSF, Servlets, EJBs, Java API para Servicios Web (JAX-WS), Arquitectura Java para Enlaces XML (JAXB), Metadatos de Servicios Web para la Plataforma Java 1.0, y muchas otras tecnologías.

Cuenta con una comunidad que es transparente en cuanto a términos de entrega de código fuente, discusiones de ingeniería, agendas, datos de descarga, etc. Dado que e Glassfish implementa la Plataforma Java Enterprise Edition (JEE) es esencialmente una forma estándar de desarrollar Aplicaciones Java Empresariales que sean portables, esto quiere decir, que puedan ser utilizadas en

más de un servidor sin importar el fabricante, sin necesidad de hacerles cambio alguno.

Existe una versión comercial denominada Oracle GlassFish Enterprise Server (antes Sun GlassFish Enterprise Server). La versión gratuita y de código libre llamada simplemente GlassFish, se distribuye bajo un licenciamiento dual a través de la licencia CDDL y la GNU GPL.

GlassFish tiene como base al servidor Sun Java System Application Server de Sun Microsystems, un derivado de Apache Tomcat.

La plataforma Java EE 6 mejora significativamente la productividad del desarrollador, presenta el perfil Web ligero para aplicaciones Web e incluye las últimas versiones de tecnologías como JAX-RS 1.1, JavaServer Faces(JSF) 2.0, Enterprise JavaBeans (EJB) 3.1, Java Persistence (App) 2.0, Context and Dependency Injection (CDI) 1.0 y mucho más. GlassFish es un servidor de aplicaciones de código abierto para Java EE.[10]

2.5.4.2 JBOSS

JBoss es un servidor de aplicaciones J2EE de código abierto implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo para el que esté disponible Java. Los principales desarrolladores trabajan para una empresa de servicios, JBoss Inc., adquirida por Red Hat en abril del 2006. El proyecto está apoyado por una red mundial de colaboradores. Los ingresos de la empresa están basados en un modelo de negocio de servicios. JBoss implementa todo el paquete de servicios de J2EE.

Es el primer servidor de aplicaciones de código abierto, preparado para la producción y certificado J2EE, disponible en el mercado, ofreciendo una plataforma de alto rendimiento para aplicaciones de e-business¹². Combinando una arquitectura orientada a servicios revolucionaria con una licencia de código

¹²e-business: consiste en introducir tecnologías de la comunicación para realizar las actividades de un negocio, Es un conjunto de nuevas tecnologías y nuevas estrategias de negocio para desarrollar estos negocios en línea.

abierto, JBoss puede ser descargado, utilizado, incrustado y distribuido sin restricciones por la licencia. Por este motivo es la plataforma más popular de middleware¹³ para desarrolladores, vendedores independientes de software y, también, para grandes empresas.

Las características destacadas de JBoss incluyen:

- Producto de licencia de código abierto sin coste adicional.
- Cumple los estándares.
- Confiable a nivel de empresa
- Incrustable, orientado a arquitectura de servicios.
- Flexibilidad consistente
- Servicios del middleware para cualquier objeto de Java
- Ayuda profesional 24x7 de la fuente
- Soporte completo para JMX[11]

2.5.4.3 TIPOS DE SERVIDORES DE APLICACIONES

Características	GLASSFISH(*)	JBoss(*)	TOMCAT(*)
Software Propietario	3	2	4
Código Abierto	8	9	7
Uso de Persistencias	8	8	3
Administración	9	8	7
Contenedor de Servlets	4	3	9
Uso Tecnologías Nuevas	9	9	3
Documentación	8	8	8
JDBC	8	8	8
Uso en Dispositivos móviles	4	8	4
Mayor crecimiento en el mercado	6	8	4

(*) Valor del puntaje en el análisis: Sobre 10 puntos.

Tabla 2-9: Servidores de Aplicaciones

Fuente: El Autor, 2013

Por las características mostradas se eligió a GlassFish como servidor de aplicación para el desarrollo del software, por su robustez, gran documentación, software libre y uso eficiente con persistencias.

¹³Middleware: software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos.

2.6 GESTOR DE BASE DE DATOS

Un gestor de base de datos o sistema de gestión de base de datos (SGBD o DBMS) es un software que permite introducir, organizar y recuperar la información de las bases de datos, en definitiva, administrarlas.

Ayuda a realizar las siguientes acciones:

- Definición de los datos
- Mantenimiento de la integridad de los datos dentro de la base de datos
- Control de la seguridad y privacidad de los datos
- Manipulación de los datos

Un sistema gestor de base de datos está compuesto de:

El gestor de base de datos.- Se trata de un conjunto de programas no visibles al usuario final que se encarga de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo.

Proporciona una interfaz entre los datos, los programas que los maneja y los usuarios finales, cualquier operación que el usuario hace contra la base de datos está controlada por el gestor.

El gestor almacena una descripción de datos lo que llamamos diccionario de datos, así como los usuarios permitidos y los permisos.

Tiene que haber un usuario administrador encargado de centralizar todas estas tareas.

Diccionario de datos.- Es una base de datos donde se guardan todas las propiedades de la base de datos, descripción de la estructura, relaciones entre los datos, etc.

El diccionario debe contener:

- La descripción externa, conceptual e interna de la base de datos
- Las restricciones sobre los datos
- El acceso a los datos
- Las descripciones de las cuentas de usuario

- Los permisos de los usuarios
- Los esquemas externos de cada programa

Administrador de la base de datos.- Es una persona o grupo de personas responsables del control del sistema de gestor de base de datos.

Las principales tareas de un administrador son:

- La definición del esquema lógico y físico de la base de datos
- La definición de las vistas de usuario
- La asignación y edición de permisos para los usuarios
- Mantenimiento y seguimiento de la seguridad en la base de datos
- Mantenimiento general del sistema gestor de base de datos.

2.6.1 POSTGRESQL

PostgreSQL es un sistema de base de datos objeto y relacional basado en postgres, versión 4.21, desarrollado en la Universidad de California en Berkeley Departamento de Informática.

Disponibles en algunas bases de datos comerciales, ya que cuentan con tipos de datos como lo son números de coma flotante, enteros, cadenas de caracteres, el dinero y las fechas. Se admite comúnmente que este modelo es inadecuado para las futuras aplicaciones de procesamiento de datos. El modelo relacional sustituido con éxito los modelos anteriores, en parte debido a su simplicidad.

PostgreSQL ofrece una potencia adicional sustancial mediante la incorporación de los conceptos adicionales siguientes de tal manera que los usuarios pueden extender fácilmente el sistema tales como:

- herencia
- Tipos de datos
- Funciones

Otras características proporcionar energía adicional y flexibilidad:

- disparadores
- reglas
- integridad transaccional

Estas características ponen PostgreSQL en la categoría de bases de datos *objeto-relacional*, siendo adecuados para apoyar a los lenguajes tradicionales de bases de datos relacionales, por lo tanto, aunque PostgreSQL tiene algunas características orientadas a objetos, es firme en el mundo de base de datos relacional. De hecho, algunas bases de datos comerciales han incorporado recientemente características por primera vez por PostgreSQL. [12]

2.6.2 ANÁLISIS COMPARATIVO DE GESTORES DE BASES DE DATOS

Si bien es cierto que MySql ofrece una mayor velocidad ya que no ocupa tantos recursos de procesador y de memoria para realizar las tareas designadas, Postgres soporta mayores transacciones para triggers y procedimientos en el servidor, pudiendo realizar varias operaciones al mismo tiempo sobre la misma tabla sin la necesidad de bloquearla.

Postgres posee una gran escalabilidad ya que es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema, haciéndole más robusto y seguro a la hora de utilizar información valiosa siendo capaz de entrar en las empresas sin muchos inconvenientes.

A continuación se detalla una tabla con los parámetros de comparación entre los gestores de base de datos antes mencionados.

Características	SQLite (*)	PostgreSQL (*)	MySQL (*)
Soporta Windows, Linux, Mac OS x, Free BSD, Solaris, Android	7	9	6
Importación y exportación de datos.	8	10	8
Soporte de interfaces c, c++, java, php, .net, odbc, jdbc	7	9	5
Control de acceso a usuarios, backup, bloqueos a nivel de registro, tablas, campos.	5	9	4
Procesamiento distribuido de transacciones	7	9	6
Definición y encriptación de datos.	5	10	7
Optimización de funciones, procedimientos almacenados, actualizaciones, eliminaciones e inserciones.	8	8	6

(*) Valor del puntaje en el análisis: Sobre 10 puntos.

Tabla 2-10 Gestores de Base de Datos más Conocidos

Fuente: El Autor, 2013

2.6.3 RESULTADO DE GESTOR DE BASE DE DATOS

Se decidió usar PostgreSQL como gestor de la base de datos del proyecto, por la carga que puede soportar al aumentar el volumen de información manejada además de la escalabilidad que posee, y la velocidad de las transacciones se ajusta al hardware del equipo. Además del factor económico ya que se cuenta con una base de datos de código abierto con licencia GPL, siendo una alternativa válida para los desarrolladores y gerentes al momento de seleccionar una herramienta informática.

Según la Tabla 2-11, Postgres posee la mayoría de las características que se necesitan para realizar el sistema cubriendo los requerimiento de almacenamiento y seguridad del gran volumen datos e información que maneja la asociación de la presidencia.

2.7 SERVICIOS

A continuación se prepara una síntesis de los servicios con los que el proyecto contará para mostrar el valor agregado que caracterizará al software desarrollado. Estos servicios son con los que cuentan las nuevas empresas hoy en día manteniendo actualizado las tecnologías que ofrece el mundo de la informática, siendo transparente para el usuario final.

2.7.1 WEB SERVICES

Los Web Services son conjuntos de protocolos y estándares que sirven para intercambiar datos entre aplicaciones, las cuales son externas siendo utilizadas para integrar las aplicaciones informáticas que están escritos en diferentes idiomas de programación y se ejecutan en diferentes plataformas.

Son muy útiles y no es necesaria la plataforma en que corran ya que lo importante es que usen el mismo lenguaje.

Un programa cliente que se conecta a un servicio Web puede leer el WSDL para determinar que funciones están disponibles en el servidor. Los tipos de datos

especiales se incluyen en el archivo WSDL en forma de XML Schema. El cliente puede usar SOAP para hacer la llamada a una de las funciones listadas en el WSDL.

El uso de los Web Service siendo una aplicación que está en internet puede acceder desde cualquier parte del mundo, manteniendo algunas restricciones por seguridad, y devuelve información similar a ejecutar una consulta en una base de datos, sin la necesidad de que los datos solicitados sean obtenidos de nuestra aplicación si no de aplicaciones externas a nivel Web, esto lo hace muy útil al no ser necesario trabajar en una misma plataforma u herramienta de programación, al ser un estándar que se lo puede crear en la mayoría de herramientas de programación.

2.7.1.1 VENTAJAS

Los Web services apuntan a ser la piedra fundamental de la nueva generación de sistemas distribuidos. Estos son algunos puntos para fundamentar esta afirmación:

- **Interoperabilidad:** Cualquier Web service puede interactuar con otro Web service. Como los Web services pueden ser implementados en cualquier lenguaje, los desarrolladores no necesitan cambiar sus ambientes de desarrollo para producir o consumir Web services.
- **Ubicuidad:** Los Web services se comunican utilizando HTTP y XML. Por lo tanto cualquier dispositivo que soporte estas tecnologías pueden implementar o acceder Web services.
- **Encapsular reduce la complejidad:** Todos los componentes en un modelo de Web services son Web service. Lo importante es la interface que el servicio provee y no como esta implementado, por lo cual la complejidad se reduce.
- **Fácil de utilizar:** El concepto detrás de los Web services es fácil de entender, incluso existen toolkits de vendedores como IBM o Microsoft que permiten a los desarrolladores crear Web services en forma rápida y fácil.

- **Soporte de la Industria:** Todas las empresas de software importantes soportan SOAP, e incluso están impulsando el desarrollo de Web services.

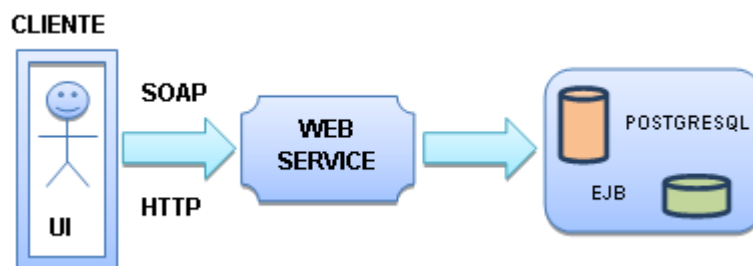


Figura 2-8 Arquitectura WEB SERVICE
Fuente: El Autor, 2013

2.7.1.2 TECNOLOGÍA WEB SERVICES

Los Web Services están contruidos con varias tecnologías que trabajan conjuntamente con los estándares que están emergiendo para asegurar la seguridad y operatividad, de modo de hacer realidad que el uso combinado de varios Web Services, independiente de la o las empresas que los proveen, este garantizado. A continuación se describen brevemente los estándares que están ocupando los Web Services.

XML

Abreviación de Extensible Markup Language. El XML es una especificación desarrollada por W3C. Permite a los desarrolladores crear sus propios tags, que les permiten habilitar definiciones, transmisiones, validaciones, e interpretación de los datos entre aplicaciones y entre organizaciones.

SOAP

Abreviación de Simple Object Access Protocol, es un protocolo de mensajería construido en XML que se usa para codificar información de los requerimientos de los Web Services y para responder los mensajes antes de enviarlos por la red. Los mensajes SOAP son independientes de los sistemas operativos y pueden ser transportados por los protocolos que funcionan en la Internet, como ser: SMTP, MIME y HTTP.

WSDL

Abreviación de Web Services Description Language, es un lenguaje especificado en XML que se ocupa para definir los Web Service como colecciones de punto de comunicación capaces de intercambiar mensajes. El WSDL es parte integral de UDDI y parte del registro global de XML, en otras palabras es un estándar de uso público (no se requiere pagar licencias ni royalties para usarlo).[13]

2.7.2 APLICACIÓN MOVIL

El desarrollo de nuevos dispositivos, en conjunto con las nuevas tecnologías trae consigo no solo el desarrollo de nuevos sistemas operativos sino un conjunto de oportunidades estratégicas para encontrar nuevos públicos. La tecnología debe comprenderse como el entorno en el que las personas invierten su atención y tiempo convirtiéndolo en el espacio correcto para desarrollar nuevas estrategias de comunicación. Los sistemas operativos Android y IOS son los líderes en el mercado teniendo en cuenta en los múltiples dispositivos en que los podemos encontrar.

2.7.2.1 SOFTWARE PARA MOVILES

A continuación se realiza un análisis de las características importantes a tener en cuenta para comenzar con el uso y desarrollo del software apropiado.

	ANDROID ^(*)	IOS ^(*)
Multitáctil	9	10
Cambio de Temas	9	5
Aplicaciones en segundo Plano	10	10
Notificaciones desde segundo plano	10	10
Función de copiar y pegar	10	10
Tienda de Aplicaciones	10	8

Sistema Operativo Abierto	10	4
Costo del equipo	5	10
Crecimiento	10	7
Uso Poblacional	7	3

(*) Valor del puntaje en el análisis: Sobre 10 puntos.

Tabla 2-11: Análisis de Software para Móviles

Fuente: El Autor, 2013

Con el análisis realizado y dependiendo los requerimientos de los usuarios, que se necesita que se acople con los recursos con los que cuentan, se ha decidido utilizar la tecnología Android para desarrollar la aplicación móvil, por estar desarrollada en Linux y ser software libre con documentación necesaria para quienes desean iniciarse en el manejo de este sistema que tiene mucha acogida a nivel mundial mostrando un crecimiento significativo y de uso personal excepcional.

2.7.2.2 ANDROID

Android, la plataforma móvil más popular del mundo, posee cientos de millones de dispositivos móviles en más de 190 países de todo el mundo. Es la mayor base instalada de cualquier plataforma móvil y de rápido crecimiento, todos los días otro millón de usuarios enciende sus dispositivos Android por primera vez y empieza a buscar aplicaciones, juegos y otros contenidos digitales.

Android es una plataforma de clase mundial para la creación de aplicaciones y juegos para los usuarios de Android en todas partes, así como un mercado abierto para la distribución a ellos al instante.

Android es un sistema operativo móvil basado en Linux, que junto con aplicaciones middleware está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google.

Las unidades vendidas de teléfonos inteligentes con Android se ubican en el primer puesto, a nivel mundial alcanzó una cuota de mercado del 50,9% más del doble que el segundo sistema operativo (iOS de Apple, Inc.)

Tiene una gran comunidad de desarrolladores escribiendo aplicaciones para extender la funcionalidad de los dispositivos. A la fecha, se han sobrepasado las 700.000 aplicaciones (de las cuales, dos tercios son gratuitas) disponibles para la tienda de aplicaciones oficial de Android: Google Play, sin tener en cuenta aplicaciones de otras tiendas no oficiales para Android, como pueden ser la App Store de Amazon o la tienda de aplicaciones Samsung Apps de Samsung. Google Play es la tienda de aplicaciones en línea administrada por Google.

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual Dalvik con compilación en tiempo de ejecución. Las bibliotecas escritas en lenguaje C incluyen un administrador de interfaz gráfica (surface manager), un framework OpenCore, una base de datos relacional SQLite, una Interfaz de programación de API gráfica OpenGL ES 2.0 3D, un motor de renderizado WebKit, un motor gráfico SGL, SSL y una biblioteca estándar de C Bionic.¹⁴

Historial de actualizaciones

Android ha visto numerosas actualizaciones desde su liberación inicial, estas actualizaciones al sistema operativo base típicamente arreglan bugs y agregan nuevas funciones, generalmente cada actualización del sistema operativo Android es desarrollada bajo un nombre en código de un elemento relacionado con postres.

Nombres de las versiones

Las versiones de Android reciben el nombre de postres en inglés. En cada versión el postre elegido empieza por una letra distinta siguiendo un orden alfabético:

- A: Apple Pie (v1.0), Tarta de manzana
- B: Banana Bread (v1.1), Pan de plátano

¹⁴<http://www.android.com/>

- C: Cupcake (v1.5), Magdalena glaseada.
- D: Donut (v1.6), Rosquilla.
- E: Éclair (v2.0/v2.1), pastel francés conocido en España como pepito, petisú, suso o canuto.
- F: Froyo (v2.2), (Abreviatura de «frozen yogurt») Yogur helado.
- G: Gingerbread (v2.3), Pan de jengibre.
- H: Honeycomb (v3.0/v3.1/v3.2), Panal de miel.
- I: Ice Cream Sandwich (v4.0), Sándwich de helado.
- J: Jelly Bean (v4.1/v4.1.2/v4.2/v4.2.1), Judía de gominola.
- K: Key Lime Pie (v4.3/5.), Tarta de lima.[14]

2.7.3 SERVICIO SMS

Los servicios que ofrecen compañías externas se encargan del envío masivo de mensajes a celulares con número que generalmente son de usuarios de una empresa determinada, para tener la información que se requiera al instante, sin la necesidad de tener conexión a internet por parte del destinatario final.

Esta forma de comunicación se convertido en un mercado potencial, ya que se puede enviar información como pancartas en Sms, publicidad en la mayoría de casos y en otros se usan para informar eventos de una empresa o accesos que se han realizado a una página de transacciones bancarias, en fin muchos usos que se le pueda dar dependiendo la necesidad que se plantee.

Las empresas que ofrecen este servicio tienen la posibilidad de conectarse mediante Web Service o mediante una interfaz Web que facilitaría el ingreso de los mensajes que se requieren ser enviados, con una contratación por paquetes de mensajes que varían desde los 100 mensajes enviados con un costo similar al envío por mensaje de las operadoras nacionales.

En nuestro caso se pretende usar este medio de comunicación para el envío masivo de las retenciones que se presentan en la asociación, de una manera ágil

y precisa, para contar instantáneamente con los detalles de los gastos que se hizo por usuario en el mes vencido.

2.8 CAPACIDAD DE ENDEUDAMIENTO SEGÚN LOSEP.

La ley Orgánica del Servidor Público, manifiesta en uno de sus artículos que a ningún servidor público se le retendrá más del 70% del valor de su sueldo, para mayor detalle vea el ANEXO A, donde indican los artículos que hacen referencia al valor de retención en este caso por parte de la Presidencia de la República del Ecuador. De acuerdo a la ley se plantea que se haga un cálculo para poder saber el máximo posible de endeudamiento de los asociados.

CAPITULO III

3 ANALISIS Y DISEÑO

3.1 PLANIFICACIÓN XP

A continuación se detallan las fases que corresponden a la metodología XP, para determinar el alcance del proyecto.

En esta fase, el cliente define lo que necesita mediante la redacción de sencillas “historias de usuarios”.

3.1.1 HISTORIAS DE USUARIOS

A continuación se citan algunas historias de usuario centradas a las sugerencias abiertas que se dieron en la Asociación de Empleados de la Presidencia de la República del Ecuador.

Número:	1	Nombre de Historia de Usuario:	Requerimientos iniciales
Referencia a la Historia de usuario número:			1
Usuario:	Carlos Franco Jeovy Toscano Luis Abalos	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Levantar la información necesaria que facilite el análisis, diseño e implementación de un sistema de automatización y control orientado a la generación de retenciones del rol de pagos con notificaciones Sms para la Asociación de Empleados de la Presidencia de la República del Ecuador desarrollado con Software Libre.</p> <p>El software debe ayudar a mejorar los procesos que se realizan en la asociación.</p> <p>El sistema debe generar reportes claros que ayuden a manejar de mejor manera los ingresos y egresos.</p> <p>Las interfaces deben ser sencillas e intuitivas.</p>		
Observaciones:	Presentar un diseño de navegabilidad.		

Tabla 3-1 Historia de Usuario: Requerimientos Iniciales

Fuente: El Autor, 2013

Número:	2	Nombre de Historia de Usuario:	Tipos de Perfiles
Referencia a la Historia de usuario número:			1
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Se requiere se controle el acceso al sistema mediante una ventana de inicio de sesión correspondiente a los diferentes tipos de perfiles de acuerdo a los usuarios.</p> <p>Se creará un perfil administrador el que será encargado de la gestión de usuarios del sistema, y el cual tendrá todos los permisos.</p> <p>Se creará un perfil secretaria el que se encargará de la gestión de información.</p> <p>Se creará un perfil asociado el cual contendrá los detalles de los descuentos de dicho asociado.</p>		
Observaciones:	Presentar un diseño de navegabilidad dependiendo los perfiles del sistema.		

Tabla 3-2 Historia de Usuario: Tipos de Perfiles

Fuente: El Autor, 2013

Número:	3	Nombre de Historia de Usuario:	Base de Datos
Referencia a la Historia de usuario número:			2
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Diseñar el modelo conceptual de la base de datos en la que se guardara la información del sistema y a la vez permita obtener acceso a información exacta y actualizada.</p> <p>El modelo deberá contener una descripción detallada de los requerimientos de información, y las descripciones de los tipos de datos, relaciones entre ellos y restricciones.</p>		
Observaciones:	El sistema ocupara como base de datos Postgresql, esto se debe tener en cuenta para definir el tipo de datos para cada uno de los campos.		

Tabla 3-3 Historia de Usuario: Base de Datos

Fuente: El Autor, 2013

Número:	4	Nombre de Historia de Usuario:	Desarrollo de inicio de sesión.
Referencia a la Historia de usuario número:			3
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media
Descripción:	<p>El sistema contendrá como pantalla de inicio un login de usuario de acuerdo a su perfil.</p> <p>La página de inicio contendrá:</p> <ul style="list-style-type: none"> • Encabezado: ícono de la asociación e imágenes de la presidencia. • Menú: en la parte izquierda de la pantalla con las opciones de acuerdo al perfil, y el nombre de la persona o perfil que se 		

	<p>registró</p> <ul style="list-style-type: none"> • Index: ¿Quiénes somos?, para todos los usuarios. • Pie de página con datos de la Universidad Politécnica Salesiana y el autor. <p>Los menús contendrán información de acuerdo a los privilegios de los usuarios, para guardar, modificar y eliminar registros.</p>
Observaciones:	El inicio de sesión debe ser validado con errores si no se encuentra el usuario, la siguiente pantalla se mostrará el nombre del usuario que inicio la sesión.

Tabla 3-4 Historia de Usuario: Desarrollo de Inicio de Sesión

Fuente: El Autor, 2013

Número:	5	Nombre de Historia de Usuario:	Gestión de Usuarios
Referencia a la Historia de usuario número:			4
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>El administrador deberá tener todos los privilegios, para poder crear, modificar y eliminar registros.</p> <p>Además de crear nuevos perfiles con sus respectivos módulos.</p> <p>Con respecto a la página para la gestión de usuarios: El administrador crear, modifica y elimina. La secretaria crea y modifica asociados no puede crear usuarios administrador ni secretaria.</p> <p>Las contraseñas no pueden ser mostradas para nadie y se creará automáticamente con el número de cédula agregado el asterisco (*), se puede resetear la contraseña de los asociados si lo solicitan a la secretaria con un botón “resetear contraseña”.</p> <p>Los usuarios se pueden dar de baja inactivándoles con un campo que mostrara si está activo o no.</p> <p>El perfil asociado puede editar su contraseña pero no la puede ver y contará con un mensaje que le ayudará a saber el grado de seguridad, este perfil no puede editar información valiosa como sueldo, darse de baja o cambiar los tipos de perfil.</p>		
Observaciones:	La validación y mensajes de error se mostrarán con mensajes que indiquen si un perfil puede o no realizar una acción.		

Tabla 3-5 Historia de Usuario: Gestión de Usuarios

Fuente: El Autor, 2013

Número:	6	Nombre de Historia de Usuario:	Menús
Referencia a la Historia de usuario número:			5
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Los menús estarán diseñados para los siguientes perfiles:</p> <p>Administrador.</p>		

	<ul style="list-style-type: none"> • Usuario. • Procesos: Bancos; Módulos; Cuentas Bancarias; Préstamos; Casa Comercial; Aportes; Cheques; Perfil; Facturas. • Retenciones: Préstamos; Facturas; Aportes; Totales. • Reportes • Opciones: Egresos; Exportar Archivo <p>Secretaria.</p> <ul style="list-style-type: none"> • Usuario. • Procesos: Préstamos; Casa Comercial; Aportes; Cheques; Facturas. • Retenciones: Préstamos; Facturas; Aportes; Totales. • Opciones: Egresos; Exportar Archivo <p>Asociado.</p> <ul style="list-style-type: none"> • Usuario. • Opciones: Retenciones Mensuales; Préstamos.
Observaciones:	De acuerdo a los perfiles se mostrarán los menús y el administrador puede dar permisos de ver o no pantallas.

Tabla 3-6 Historia de Usuario: Menús
Fuente: El Autor, 2013

Número:	7	Nombre de Historia de Usuario:	Páginas Estándares Listar
Referencia a la Historia de usuario número:			6
Usuario:	Carlos Franco Jeovany Toscano Luis Abalos	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Las pantallas tendrán un formato estándar y se busca que sea similares unas de las otras para que sea de fácil manejo, y se acople con las necesidades requeridas.</p> <p>La parte del contenido de la página Web, deberá mostrar un rótulo de la página que se está mostrando, un panel de acciones con íconos que permitan crear nuevo, dependiendo en la opción que se encuentre. Y en la parte derecha de este panel íconos para regresar a la página principal (home) y salir del sistema.</p> <p>Para cada acción opción del menú se mostrarán en forma de listas que corresponden al ítem seleccionado, esta tabla de listas mostrará información necesaria que ayude a identificar con brevedad un registro. La lista contendrá filtros para búsqueda y para ordenar los resultados, en la parte derecha se podrá seleccionar el registro y editarlo con un clic en un botón o imagen.</p>		
Observaciones:			

Tabla 3-7 Historia de Usuario: Páginas Estándares Listar
Fuente: El Autor, 2013

Número:	8	Nombre de Historia de Usuario:	Páginas Estándares Editar
Referencia a la Historia de usuario número:			7
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Las pantallas tendrán un formato estándar.</p> <p>La parte del contenido de la página Web, deberá mostrar un rótulo de la página que se está mostrando, un panel de acciones con íconos que permitan crear nuevo, actualizar, eliminar. Y en la parte derecha de este panel íconos para regresar a la página principal (home) y salir del sistema.</p> <p>Las validaciones en la página serán similares a las necesarias para crear nuevo registro. Los mensajes se desplegarán en la parte superior de la pantalla, de color rojo para errores y de color azul para información.</p>		
Observaciones:			

Tabla 3-8 Historia de Usuario: Páginas Estándares Editar
Fuente: El Autor, 2013

Número:	9	Nombre de Historia de Usuario:	Páginas Estándares Crear Nuevo
Referencia a la Historia de usuario número:			8
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Las pantallas tendrán un formato estándar.</p> <p>La parte del contenido de la página Web, deberá mostrar un rótulo de la página que se está mostrando, un panel de acciones con íconos que permitan crear nuevo, guardar, limpiar. Y en la parte derecha de este panel íconos para regresar a la página principal (home) y salir del sistema.</p> <p>Contendrán información necesaria para llenar los campos de las tabas y sus respectivas relaciones, validando la información. Los campos que son claves principales se generan automáticamente mediante un método. Los mensajes se desplegarán en la parte superior de la pantalla, de color rojo para errores y de color azul para información.</p>		
Observaciones:	La selección de un usuario en las pantallas que así lo requieren se filtrará por apellidos seguidos por los nombres.		

Tabla 3-9 Historia de Usuario: Páginas Estándares Crear Nuevo
Fuente: El Autor, 2013

Número:	10	Nombre de Historia de Usuario:	Opción Retenciones
Referencia a la Historia de usuario número:			9
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta

Descripción:	<p>Páginas estándares retenciones de: Préstamos y Aportes.</p> <p>El panel de acciones debe tener íconos: crear nueva retención y eliminar retenciones, solo para el perfil administrador.</p> <p>Se podrán ejecutar búsquedas de una determinada fecha la cual será escogida desde una lista desplegable, que estará vinculada con las retenciones creadas y almacenadas en la respectiva tabla.</p> <p>Luego de seleccionar la fecha y presionar el botón buscar se mostrará una tabla con la información correspondiente a la búsqueda, además podrá ser filtrada y ordenada de acuerdo a los requerimientos y editar los registros siempre y cuando la fecha de creación del registro no supere los 15 días de la fecha actual, de esta forma se controlará que no se puedan modificar registros con fechas que ya hayan sido descontadas.</p> <p>El botón eliminar permitirá seleccionar de una lista de fechas a ser eliminadas.</p> <p>Los mensajes se desplegarán en la parte superior de la pantalla, de color rojo para errores y de color azul para información.</p>
Observaciones:	Crear retención y eliminar retención solo son acciones que un administrador pueda realizar.

Tabla 3-10 Historia de Usuario: Opción Retenciones
Fuente: El Autor, 2013

Número:	11	Nombre de Historia de Usuario:	Retenciones de Factura
Referencia a la Historia de usuario número:			10
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Al dar clic en la opción de retenciones de facturas, mostrará en una lista los registros ingresados para el mes en curso, permitirá ordenar la información y búsquedas por filtros.</p> <p>El panel de acciones debe tener íconos: crear nueva retención y buscar retenciones.</p> <p>El botón crear nueva retención, la cual vendría a ser el descuento de una casa comercial para un asociado, se mostrará un menú con la factura de la casa comercial en una lista desplegable, los asociados ordenados por apellido, el monto de la retención y la fecha cuando se le genera.</p> <p>Al guardar se visualizará en una lista el último registro que se almacenó para tener un mejor control de la información y saber que a ese usuario se le han hecho varios descuentos de una misma casa comercial en ese mes.</p> <p>El botón buscar retenciones de facturas, deberá contener dos campos completados con fechas desde y hasta, para tener el rango de la fecha de corte que se realizó la retención.</p> <p>Los mensajes se desplegarán en la parte superior de la pantalla, de color rojo para errores y de color azul para información.</p>		

Observaciones:	
-----------------------	--

Tabla 3-11 Historia de Usuario: Retenciones Factura
Fuente: El Autor, 2013

Número:	12	Nombre de Historia de Usuario:	Retenciones Totales
Referencia a la Historia de usuario número:			11
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Al dar clic en la opción de retenciones totales, mostrará un menú el cual permitirá generar las retenciones que será el valor de descuento al asociado, se deberá seleccionar las fechas que corresponden al corte del descuento, y un botón generar.</p> <p>Validación:</p> <ul style="list-style-type: none"> • No permitirán campos vacíos o que no sean fechas, para esto se desplegará un calendario que facilitará la selección. • El campo fecha_hasta deber ser mayor al campo fecha_desde. • El sistema no permitirá generar más de una vez las retenciones totales. <p>El panel de acciones debe tener íconos: Generar retenciones (perfil secretaria) Nueva retención, Eliminar retenciones y buscar (perfil administrador)</p> <p>La opción botón buscar retenciones totales, mostrará una lista desplegable de las fechas que se han realizado las retenciones, al seleccionar y dar clic en buscar aparecerá una pantalla con la información requerida, esta se podrá editar si la fecha no supera a los 15 días desde que se efectuó la retención comparada con la fecha actual.</p> <p>Los mensajes se desplegarán en la parte superior de la pantalla, de color rojo para errores y de color azul para información.</p>		
Observaciones:			

Tabla 3-12 Historia de Usuario: Retenciones Totales
Fuente: El Autor, 2013

Número:	13	Nombre de Historia de Usuario:	Reportes
Referencia a la Historia de usuario número:			12
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media
Descripción:	<p>La opción reportes, desplegará un menú que permite seleccionar la fecha de una lista desplegable que contendrá las fechas correspondientes a las retenciones totales, al presionar esta búsqueda nos mostrará los valores totales detallados por:</p> <ul style="list-style-type: none"> • Aportes generados. • Casas comerciales. • Prestamos debitados. <p>En el menú de acciones estará un botón que permitirá imprimir el reporte mostrado.</p>		

Observaciones:	La información mostrada será capaz ordenarse y realizar búsquedas por filtro.
-----------------------	---

Tabla 3-13 Historia de Usuario: Reportes
Fuente: El Autor, 2013

Número:	14	Nombre de Historia de Usuario:	Archivo Retenciones
Referencia a la Historia de usuario número:			13
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media
Descripción:	<p>La opción exportar archivo, mostrará un menú que permite seleccionar la fecha de una lista desplegable que contendrá las fechas correspondientes a las retenciones totales, al presionar esta búsqueda nos mostrará en una tabla los valores correspondientes al descuento que se le retendrá a cada asociado.</p> <ul style="list-style-type: none"> • Aportes generados. • Casas comerciales. • Prestamos debitados. <p>En el menú de acciones estará una opción que permitirá exportar la información mostrada a:</p> <ul style="list-style-type: none"> • Pdf. • Excel. • Csv • Xml. <p>Esta información es la que se envía a la Presidencia de la República del Ecuador para que haga los descuentos, en una especie de factura que entrega la asociación.</p>		
Observaciones:	La información mostrada será capaz ordenarse y realizar búsquedas por filtro.		

Tabla 3-14 Historia de Usuario: Archivo Retenciones
Fuente: El Autor, 2013

Número:	15	Nombre de Historia de Usuario:	Web Service
Referencia a la Historia de usuario número:			14
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media
Descripción:	<p>Este método que está en la Web permitirá que el área financiera de la Presidencia de la república del Ecuador se pueda conectar para obtener las retenciones mensuales de los asociados, con este Web service se pretende usar de una manera más eficiente la información que se generaba como archivo en la historia de usuario número: 14.</p> <p>El wsdl será: /ServiceMensualService/ServiceMensual?WSDL. Método: buscarRetenciones.</p> <p>Se deberá ingresar la fecha en que el sistema le devuelve las retenciones totales de los asociados, la fecha debe ser en el siguiente formato: dd/mm/yyyy</p>		

Observaciones:	

Tabla 3-15 Historia de Usuario: Web Service
Fuente: El Autor, 2013

Número:	16	Nombre de Historia de Usuario:	Corrección Listar Facturas								
Referencia a la Historia de usuario número:			7								
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media								
Descripción:	<p>En la opción: Procesos / Facturas</p> <p>Se listan todas las facturas que se han generado, pero para detallar de mejor manera, se mostrará en una tabla todo el detalle de la factura.</p> <p>A la tabla: tb_factura se requiere agregar los siguientes campos:</p> <table><tr><td>FechaFactura:</td><td>Ruc:</td></tr><tr><td>Factura.No:</td><td>autorización:</td></tr><tr><td>Sub:</td><td>IVA:</td></tr><tr><td>Porcentaje:</td><td>MontoTotal:</td></tr></table>			FechaFactura:	Ruc:	Factura.No:	autorización:	Sub:	IVA:	Porcentaje:	MontoTotal:
FechaFactura:	Ruc:										
Factura.No:	autorización:										
Sub:	IVA:										
Porcentaje:	MontoTotal:										
Observaciones:	EL campo porcentaje se lo usará para almacenar el valor total de facturas que no generan iva como las casas de salud en general.										

Tabla 3-16 Historia de Usuario: Corrección Listar Facturas
Fuente: El Autor, 2013

Número:	17	Nombre de Historia de Usuario:	Corrección Egresos
Referencia a la Historia de usuario número:			16
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media
Descripción:	<p>Se debe controlar que hay usuarios que ya no pertenecen a la asociación, pero aún se les sigue descontando, estos hacen su aporte por medio de pago directo a la asociación y debería haber un registro para estos usuarios.</p> <p>Las facturas que se ingresan deben tener campos obligatorios como la fecha en que se registró la factura al sistema.</p>		
Observaciones:			

Tabla 3-17 Historia de Usuario: Corrección Egresos
Fuente: El Autor, 2013

Número:	18	Nombre de Historia de Usuario:	Envío de Sms
Referencia a la Historia de usuario número:			13
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Media
Descripción:	El sistema contará con una opción que permita enviar Sms a todos los números de celular de los asociados con la descripción básica del		

	descuento para el mes en curso.
Observaciones:	Validar que el mensaje de texto este en el rango de 150 caracteres.

Tabla 3-18 Historia de Usuario: Envío de sms
Fuente: El Autor, 2013

Número:	19	Nombre de Historia de Usuario:	Menú Asociado1
Referencia a la Historia de usuario número:			13
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Al ingresar al sistema mostrará en el menú sus apellidos y nombres.</p> <p>El menú contendrá 2 ítems.</p> <p>1) Usuario: En esta opción se podrá observar y editar la información de su propio usuario, pero no información importante que permite generar los descuentos.</p> <p>Además de poder modificar la contraseña, este campo solo se muestra para el asociado e indica el grado de seguridad de la contraseña escrita.</p> <p>En el menú de acciones se ubicará un ícono para actualizar los datos.</p>		
Observaciones:			

Tabla 3-19 Historia de Usuario: Menú Asociado1
Fuente: El Autor, 2013

Número:	20	Nombre de Historia de Usuario:	Menú Asociado2
Referencia a la Historia de usuario número:			13
Usuario:	Carlos Franco	Prioridad en negocio: (Alta/Media/Baja)	Alta
Descripción:	<p>Al ingresar al sistema mostrara en el menú sus apellidos y nombres.</p> <p>El menú contendrá 2 ítems.</p> <p>1) Usuario:</p> <p>2) Opciones:</p> <p>Prestamos: detallará los préstamos que tiene con la asociación indicando las cuotas restantes que tiene que pagar.</p> <p>En el menú de acciones se ubicará un ícono para imprimir esta información.</p> <p>Retenciones Mensuales: mostrará una lista con todas las retenciones que han sido generadas para ese usuario con una descripción de la misma.</p> <p>En el menú de acciones estará una opción que permitirá exportar e imprimir la información mostrada.</p> <ul style="list-style-type: none"> • Pdf. • Excel. • Xml. • Imprimir <p>Al final del registro habrá los siguientes íconos:</p> <ul style="list-style-type: none"> • Ver: mostrará la información detallada de la retención 		

	<p>seleccionada, agrupada por descuentos de aportes, préstamos y casas comerciales.</p> <p>En el menú de acciones se ubicará un ícono para imprimir esta información.</p> <ul style="list-style-type: none"> • Imprimir: mostrará la información de la retención seleccionada en una sola tabla la cual se desplegará tipo factura con el detalle, su valor y al final el total de la suma de las retenciones <p>En el menú de acciones estará una opción que permitirá exportar e imprimir la información mostrada.</p> <ul style="list-style-type: none"> ○ Pdf. ○ Excel. ○ Csv ○ Xml. ○ Imprimir
Observaciones:	

Tabla 3-20 Historia de Usuario: Menú Asociado2

Fuente: El Autor, 2013

3.1.2 TARJETAS CRC

Suponen una forma de pensar propia de la orientación a objetos que rompe con el diseño tradicional. Cada tarjeta representa un objeto. En la cabecera se pone el nombre de la clase a la que pertenece. Las responsabilidades se ponen en la parte izquierda y las clases con las que colabora a la parte derecha de cada responsabilidad. Lo que se hace en una sesión es simular el funcionamiento del sistema a mano.

Sesiones	
Usuario Contraseña Autenticar Ingresar	

Tabla 3-21 Tarjeta Crc: Sesiones

Fuente: El Autor, 2013

Usuarios	
Información básica de Usuario Validar Guardar Buscar Listar	Sesiones

Tabla 3-22 Tarjeta Crc: Usuarios

Fuente: El Autor, 2013

Bancos	
Nombre del banco Guardar Buscar Listar	Perfil

Tabla 3-23 Tarjeta Crc: Bancos
Fuente: El Autor, 2013

Cheques	
Nombre del banco Número de cheque Fecha de emisión Valor Guardar Buscar Listar	Bancos

Tabla 3-24 Tarjeta Crc: Cheques
Fuente: El Autor, 2013

Casas Comerciales	
Nombre casa comercial Información de contacto Guardar Buscar Listar	Perfil

Tabla 3-25 Tarjeta Crc: Casas Comerciales
Fuente: El Autor, 2013

Facturas	
Tomar información de casa comercial Tomar número de cheque Fecha Información de factura Total a pagar Guardar Buscar Listar	Casas Comerciales Cheques

Tabla 3-26 Tarjeta Crc: Facturas
Fuente: El Autor, 2013

Préstamos	
Tomar información de usuario Tomar número de cheque Cuotas Fecha Guardar	Usuarios Cheques

Buscar	
Listar	

Tabla 3-27 Tarjeta Crc: Préstamos

Fuente: El Autor, 2013

Aportes	
Nombre de aporte	Perfil
Monto	
Nombramiento	
Activo	
Detalle	
Meses de aporte	
Guardar	
Buscar	
Listar	

Tabla 3-28 Tarjeta Crc: Aportes

Fuente: El Autor, 2013

Retenciones de Prestamos	
Toma información de usuarios	Prestamos Usuarios
Toma información de prestamos	
Monto	
Fecha	
Guardar	
Buscar	
Listar	
Generar	
Editar	

Tabla 3-29 Tarjeta Crc: Prestamos

Fuente: El Autor, 2013

Retenciones de Aportes	
Toma información de usuarios	Aportes Usuarios
Toma información de aportes	
Monto	
Fecha	
Guardar	
Buscar	
Listar	
Generar	
Editar	

Tabla 3-30 Tarjeta Crc: Aportes

Fuente: El Autor, 2013

Retenciones de Facturas	
Toma información de usuarios	Facturas Usuarios

Toma información de facturas Monto Fecha Guardar Buscar Listar Generar Editar	
---	--

Tabla 3-31 Tarjeta Crc: Facturas
Fuente: El Autor, 2013

Retenciones Totales	
Toma información de: Facturas, Aportes, Préstamos Fecha Desde Fecha Hasta Generar Buscar Listar Editar	Retenciones de Facturas Retenciones de Aportes Retenciones de Préstamos

Tabla 3-32 Tarjeta Crc: Retenciones Totales
Fuente: El Autor, 2013

Exportar Archivo	
Seleccionar fecha Buscar Exportar	Retenciones Totales

Tabla 3-33 Tarjeta Crc: Exportar Archivo
Fuente: El Autor, 2013

Web Service	
Seleccionar fecha Buscar	Retenciones Totales

Tabla 3-34 Tarjeta Crc: Web Service
Fuente: El Autor, 2013

Cuentas Asociación	
Tipo Monto Fecha Guardar Limpiar Listar Editar	Perfil

Tabla 3-35 Tarjeta Crc: Cuentas Asociación
Fuente: El Autor, 2013

3.1.3 RESTRICCIONES

A continuación se hace énfasis a las restricciones que se implementarán en el sistema. Por ejemplo, un usuario solo podrá ejecutar las tareas previstas en su perfil, las retenciones totales solo son generadas una cada mes, con una fecha establecida que servirá para ejecutar búsquedas y exportar archivos.

3.1.3.1 RESTRICCIONES DE INTERFACES DE USUARIO

La aplicación desarrollada, está diseñada de manera que se rija a estándares y normas GUI¹⁵ tanto en sus páginas como ventanas para el uso adecuado de los diferentes usuarios, sabiendo que éstas normas están plenamente dedicadas a aquellos usuarios con conocimientos básicos y medios sobre Informática, y también sobre el uso de recursos informáticos como Internet.

El medio con que el usuario puede comunicarse con una máquina o una computadora, comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar, las interfaces básicas de usuario son aquellas que incluyen elementos como menús, ventanas, teclado y ratón, en general, todos aquellos canales por los cuales se permite la comunicación entre el ser humano y la computadora.

Dentro de la interfaz de usuario se puede definir tres puntos básicos.

- El primero se basa en la interacción con el usuario, intenta buscar una armonía comunicacional con los usuarios.
- El segundo, optimiza la forma en la que se le presentan los datos al usuario y crea un entorno de trabajo sencillo, amigable y fácil de comprender.
- Y tercero, tenemos la estructura del sistema, donde se perfecciona estructuralmente la interfaz gráfica; de esta forma, compromete al entorno

¹⁵ GUI-Interfaz gráfica de Usuario.

GUI en la navegabilidad y en las funciones operacionales entre una sección y la otra del sistema.

3.1.3.2 RESTRICCIONES DE INTERFACES DE COMUNICACIONES

La aplicación al ser desarrollada deberá contar con ciertos requerimientos que se deben definir en un inicio para no tener problemas en la posterioridad, es necesario que la estación de trabajo o dispositivo móvil que se utilice tenga acceso a internet, o este configurado dentro de la red donde se encuentra el servidor de aplicaciones para que se visualice en un entorno Web, además de estar publicada correctamente en una intranet o dominio.

En conclusión, se deberá recurrir a la utilización de una buena infraestructura.

Navegadores Soportados

Concibiendo la idea de que la aplicación será montada y ejecutada en plataformas de software libre, se eligió como navegador recomendable a Mozilla Firefox 17.0.1 o superior, el cual viene en la mayoría de distribuciones que Linux posee pero en versiones anteriores a la recomendada la cual es de fácil actualización si se requiere, además posee versiones gratuitas de libre acceso para sistemas operativos de Microsoft como lo es Windows.

Versiones del Sistema Android

Las versiones de Android reciben el nombre de postres en inglés. En cada versión el postre elegido empieza por una letra distinta siguiendo un orden alfabético, la versión recomendada para usar la aplicación correctamente es: Ice Cream Sandwich (v4.0), Sándwich de helado.

3.1.3.3 RESTRICCIONES DE SOFTWARE

Dando cumplimiento al decreto presidencial 1014, que establece como política gubernamental para las Entidades de la Administración Pública Central, la

utilización de Software Libre en sus sistemas y equipamientos informáticos, las herramientas de desarrollo que se utilizarán, se detallan a continuación:

- JSF
- GlassFish.
- PostgreSQL.
- WebMail.
- Servidor FTP.
- Chat Interno.
- Android

3.1.4 REQUERIMIENTOS DEL SISTEMA

En base a la evaluación de la tecnología existente en la asociación, se determinó que los componentes técnicos que se posee permiten hacer uso de los mismos en la puesta a producción del sistema propuesto

Servidor.

En cuanto a Hardware, donde se instalará el sistema planteado el servidor consta de las siguientes características:

- COMPAQ ProLiant DL360
 - Procesador Intel(R) Pentium(R) III CPU 1.26 Ghz
 - Memoria RAM 1.00 GB
 - Disco Duro 128 GB
 - Disco Duro externo 1 TB
 - Unidad DVD-RW
 - Tarjeta de Red
 - Monitor
 - Mouse
 - Teclado
 - UPS

Estación de Trabajo.

Las estaciones de trabajo donde se podrá acceder a la aplicación se requieren que cuenten con conexión a internet mediante un browser, preferiblemente la última versión de Mozilla Firefox.

3.2 DISEÑO DEL SISTEMA

3.2.1 DIAGRAMAS DE CASOS DE USO

A continuación se detallan los diagramas de casos de uso que muestra los diferentes procesos que realizan los diferentes usuarios que tiene el sistema desde la creación de usuarios hasta la generación de reportes.

3.2.1.1 DIAGRAMA DE INICIO DE SESIÓN DE USUARIOS.

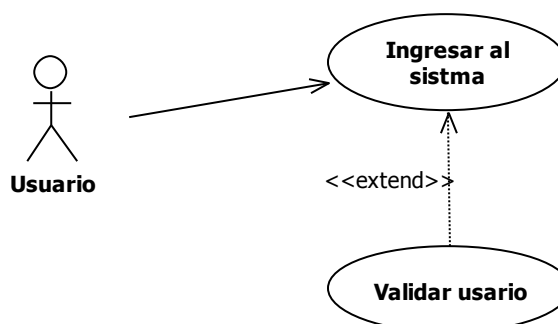


Figura 3-1 Diagrama Inicio de Sesión de Usuarios.
Fuente: El Autor, 2013

Especificación de caso de uso: inicio de sesión de usuarios.

Caso de uso	Inicio de sesión de usuarios
Objetivo	Crear la interfaz de inicio de sesión de usuarios para ingresar al sistema.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> Los usuarios que inicien sesión debes estar previamente registrados con su respectivo nombre de inicio de sesión y contraseña. El usuario debe tener conocimiento de sus datos para la

	autenticación en el sistema usuario y contraseña.
Acciones Básicas	<ul style="list-style-type: none"> • Ingresar al portal Web de la asociación. • Ingresar el nombre de inicio de sesión y la contraseña. • El sistema valida los datos mediante la comprobación de la contraseña y usuario en la base de datos. • El usuario finalmente ha ingresado al sistema según su perfil.

Tabla 3-36 Caso de Uso Inicio de Sesión de Usuarios.

Fuente: El Autor, 2013

3.2.1.2 DIAGRAMA DE CREACIÓN DE USUARIOS.

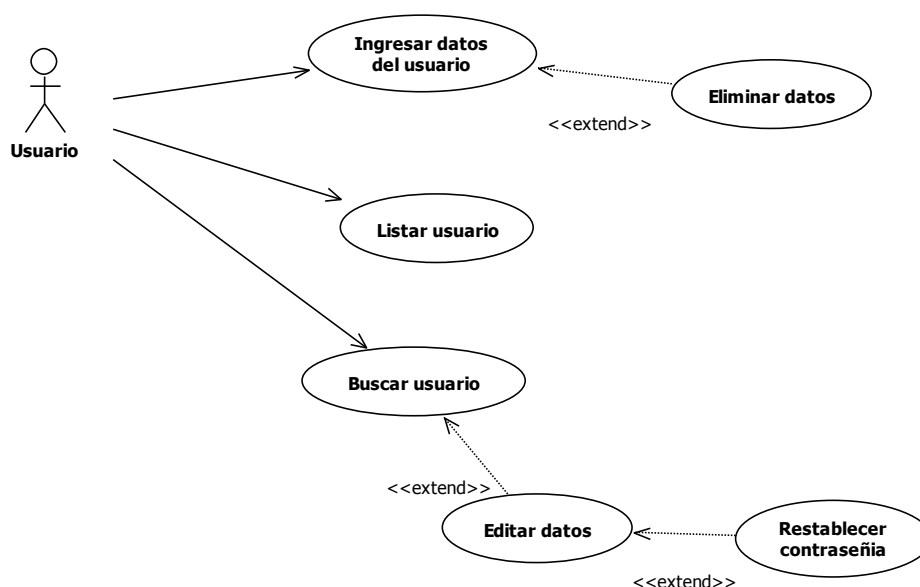


Figura 3-2 Diagrama Creación de Usuarios

Fuente: El Autor, 2013

Especificación de caso de uso: creación de usuarios.

Caso de uso	Creación de usuarios
Objetivo	Creación de usuarios por medio de la interfaz Usuario.
Actores	Administrador, Secretaria
Precondiciones	<ul style="list-style-type: none"> • Ser empleado de la Presidencia de la República del Ecuador. • Pertenecer a la asociación de la Presidencia de la República del Ecuador.
Acciones Básicas	<ul style="list-style-type: none"> • Ingresar al portal Web. • El usuario debe ingresar los datos personales del nuevo asociado y el perfil que le corresponda. • El sistema le provee de una contraseña transitoria.

	<ul style="list-style-type: none"> • Se muestran los datos informativos con el nombre de usuario y la contraseña que se encuentra oculta. • Puede editar la información del usuario ingresado. • No se puede editar la contraseña ya que se encuentra oculta, y es de manejo personal • El administrador o la secretaria pueden restablecer la contraseña.
--	--

Tabla 3-37 Caso de Uso Creación de Usuarios.

Fuente: El Autor, 2013

3.2.1.3 DIAGRAMA DE INGRESO DE CASAS COMERCIALES.

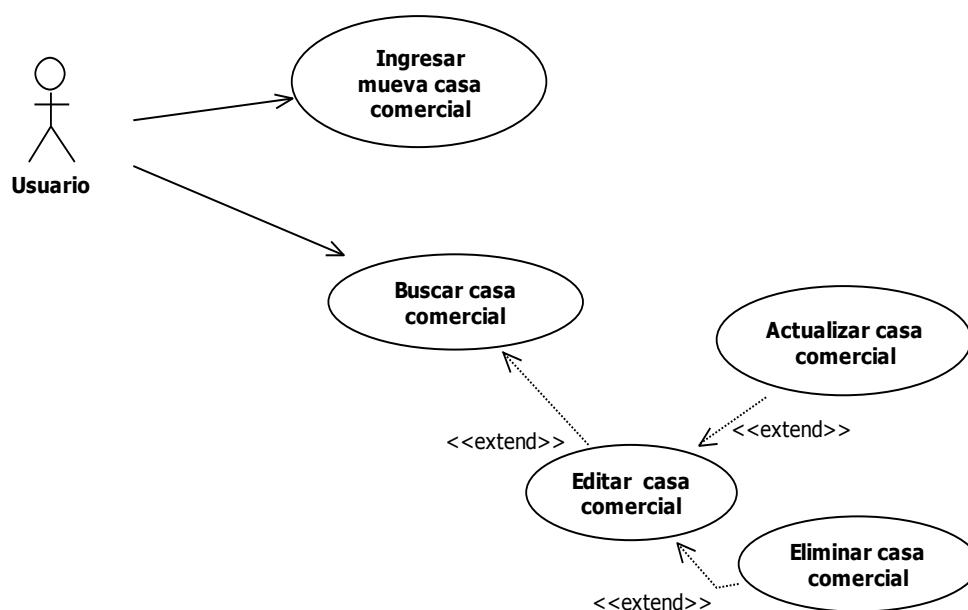


Figura 3-3 Diagrama Ingreso de Casas Comerciales.

Fuente: El Autor, 2013

Especificación caso de uso: ingreso de casas comerciales.

Caso de uso	Ingreso de casas comerciales
Objetivo	Crear la interfaz de ingreso de nuevas casa comerciales con sus respectivos datos informativos.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> • La asociación de la presidencia de la república debe ser cliente de la casa comercial. • La casa comercial debe otórgale un convenio a la asociación.
Acciones Básicas	<ul style="list-style-type: none"> • Registrar los datos informativos de la casa comercial.

	<ul style="list-style-type: none"> • Buscar casa comercial. • Editar datos de la casa comercial. • Actualizar datos de la casa comercial. • Eliminar casa comercial.
--	--

Tabla 3-38 Caso de Uso Ingreso de Casas Comerciales.

Fuente: El Autor, 2013

3.2.1.4 DIAGRAMA DE REGISTRO DE BANCOS.

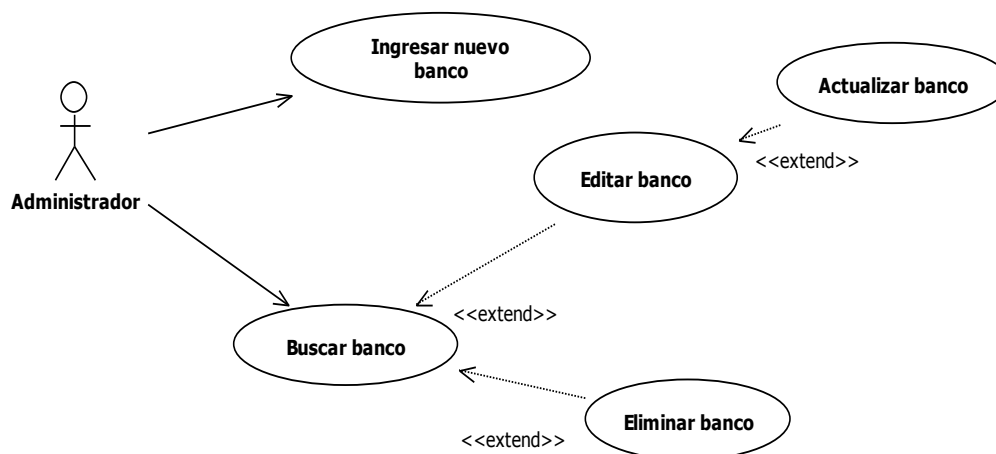


Figura 3-4 Registro de Bancos

Fuente: El Autor, 2013

Especificación caso de uso: registro de bancos.

Caso de uso	Registro de bancos
Objetivo	Crear la interfaz del registro de nuevos bancos.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> • La asociación ha convenido contar con los servicios del banco.
Acciones Básicas	<ul style="list-style-type: none"> • Registrar los datos informativos del banco. • Editar nombre del banco. • Eliminar banco.

Tabla 3-39 Caso de Uso Registro de Bancos

Fuente: El Autor, 2013

3.2.1.5 DIAGRAMA DE REGISTRO DE CUENTA BANCARIA.

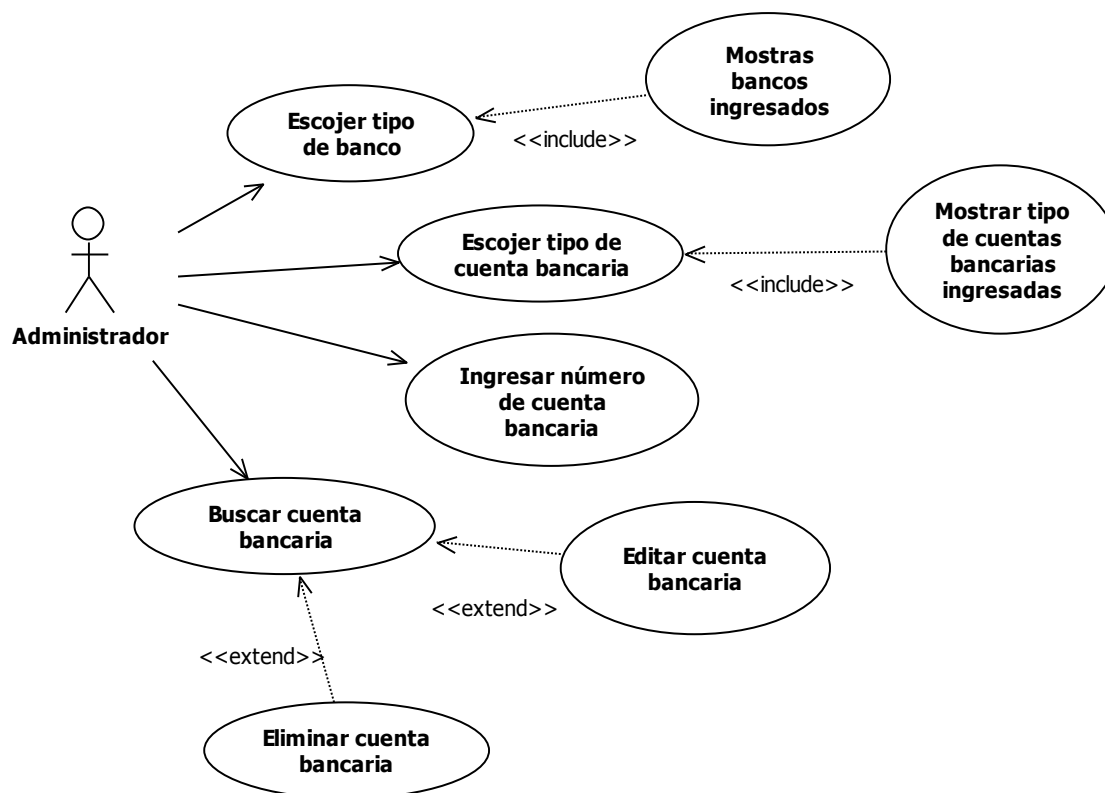


Figura 3-5 Diagrama Registro de Cuenta Bancaria
Fuente: El Autor, 2013

Especificación caso de uso: registro de cuenta bancaria.

Caso de uso	Registro de cuenta bancaria
Objetivo	Crear la interfaz del registro de nuevas cuentas bancarias.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> La asociación ha convenido contar con los servicios de una cuenta bancaria. Si la entidad bancaria con la que ha convenido tener una cuenta bancaria es nueva, ingresar primero los datos del nuevo banco. Determinar los tipos de cuentas bancarias con los que cuenta la asociación.
Acciones Básicas	<ul style="list-style-type: none"> Seleccionar el banco donde se ha creado la cuenta bancaria. Escoger el tipo de cuenta bancaria.

	<ul style="list-style-type: none"> • Registrar el número de la cuenta bancaria que se ha creado. • Editar datos de la cuenta bancaria. • Eliminar datos de la cuenta bancaria.
--	---

Tabla 3-40 Caso de Uso Registro de Cuenta Bancaria

Fuente: El Autor, 2013

3.2.1.6 DIAGRAMA DE INICIO DE INGRESO DE CHEQUES.

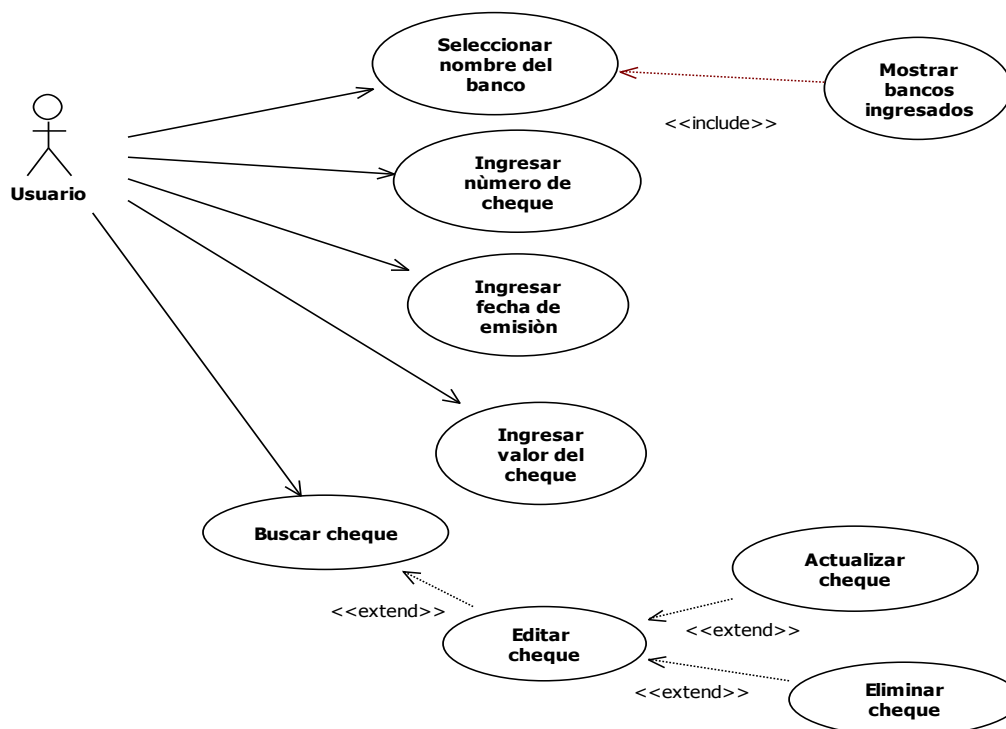


Figura 3-6 Diagrama Ingreso de Cheques.

Fuente: El Autor, 2013

Especificación caso de uso: ingreso de cheques.

Caso de uso	Ingreso de cheques.
Objetivo	<ul style="list-style-type: none"> • Crear la interfaz de ingreso de cheques. • Registrar cada uno de los cheques con la información correspondiente de pagos a las casas comerciales o los pagos de los préstamos que realizan los asociados.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> • La asociación ha convenido contar con los servicios de giros de cheques de un banco. • Si la entidad bancaria con la que se va a girar los cheques es nueva, ingresar primero los datos del nuevo banco.

Acciones Básicas	<ul style="list-style-type: none"> • Seleccionar el nombre del banco con el que se va a girar los cheques. • Ingresar el número de cheque. • Buscar cheque. • Editar cheque. • Actualizar cheque. • Eliminar cheque.
-------------------------	--

Tabla 3-41 Caso de Uso Ingreso de Cheques.
Fuente: El Autor, 2013

3.2.1.7 DIAGRAMA DE REGISTRO DE FACTURAS.

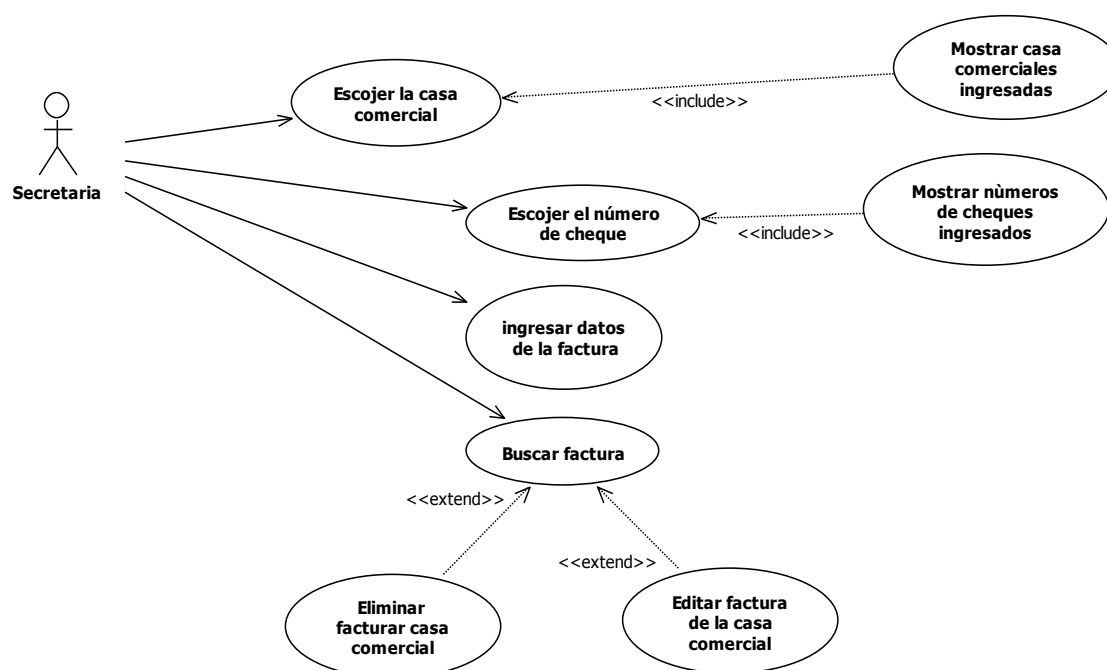


Figura 3-7 Diagrama Registro de Facturas.
Fuente: El Autor, 2013

Especificación caso de uso: registro de facturas.

Caso de uso	Registro de facturas
Objetivo	Crear la interfaz de registro de facturas de las casas comerciales con sus respectivos datos informativos.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> • El administrador debe ingresar al sistema los cheques con los cuales se le va a cancelar la factura. • La secretaria previamente debe ingresar la casa comercial que va a ser facturada.

Acciones Básicas	<ul style="list-style-type: none"> Registrar la factura de la casa comercial con el valor a cancelar, descuento a la asociación y valor total de gastos que los asociados han realizado.
-------------------------	---

Tabla 3-42 Caso de Uso Registro de Facturas.

Fuente: El Autor, 2013

3.2.1.8 DIAGRAMA DE INGRESO DE PRÉSTAMOS.

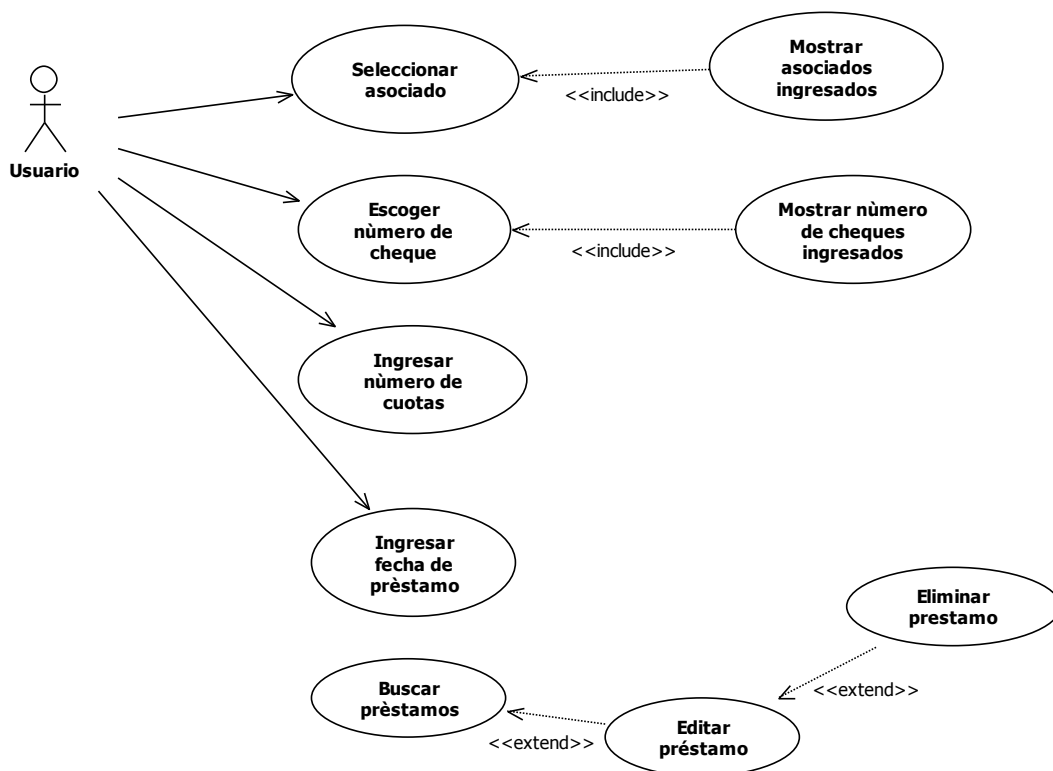


Figura 3-8 Diagrama Ingreso de Préstamos.

Fuente: El Autor, 2013

Especificación caso de uso: ingreso de préstamos.

Caso de uso	Ingreso de préstamos.
Objetivo	Crear la interfaz de ingreso de préstamos que la asociación pone a su disposición a sus asociados mediante previos requisitos que deben cumplirlos.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> El asociado debe tener los suficientes ingresos para que la asociación le conceda el préstamo. El asociado debe tener al día los pagos obligatorios mensuales que recibe la asociación.

Acciones Básicas	<ul style="list-style-type: none"> • Registrar los préstamos por cada asociado.
-------------------------	--

Tabla 3-43 Caso de Uso Ingreso de Préstamos.

Fuente: El Autor, 2013

3.2.1.9 DIAGRAMA DE INGRESO DE APORTES.

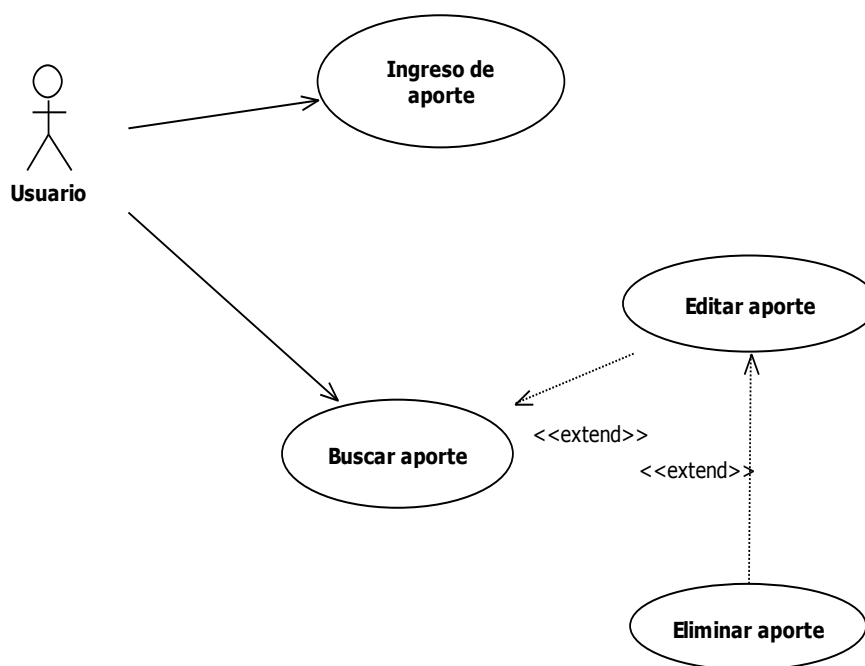


Figura 3-9 Diagrama Ingreso de Aportes.

Fuente: El Autor, 2013

Especificación caso de uso: ingreso de aportes.

Caso de uso	Ingreso de aportes
Objetivo	Crear la interfaz de ingreso de aportes que la asociación ha predispuesto a sus asociados los cuales son generados mensualmente.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> • Ser asociado a la presidencia de la república del Ecuador. • El asociado debe estar activo. • La retención depende si el asociado tiene o no nombramiento.
Acciones Básicas	<ul style="list-style-type: none"> • Registrar los aportes que se generarán mensualmente.

Tabla 3-44 Caso de Uso Ingreso de Aportes.

Fuente: El Autor, 2013

3.2.1.10 DIAGRAMA DE GENERACIÓN DE RETENCIÓN DE PRÉSTAMOS.

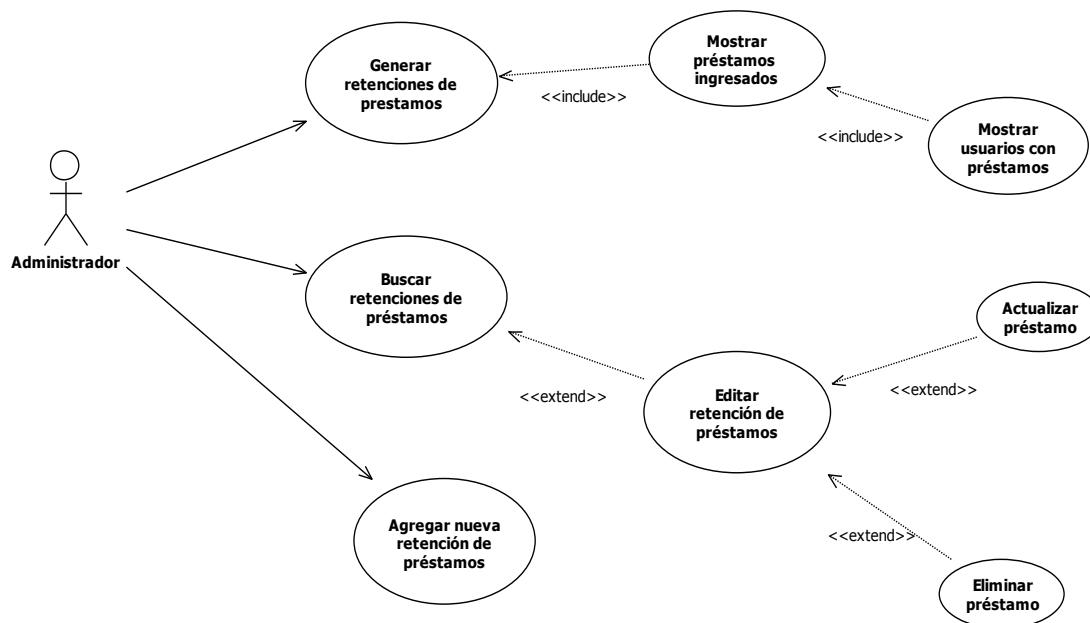


Figura 3-10 Diagrama Generación de Retención de Préstamos.
Fuente: El Autor, 2013

Especificación caso de uso: generación de retención de préstamos.

Caso de uso por mes	Generación de Retención de Préstamos.
Objetivo	Crear la interfaz de Generación de Retención de Préstamos para saber cuánto se le debe descontar al asociado de los préstamos que se le ha otorgado.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> Se debe ingresar préstamos en el menú de préstamos. El préstamo debe encontrarse activo. El número de cuotas debe ser mayor a cero.
Acciones Básicas	<ul style="list-style-type: none"> Generar retenciones de préstamos La generación de retenciones de préstamos se realiza automáticamente. El resultado de esta generación es una lista retenciones que se detallan con el código de la retención del préstamo, código del usuario, código del préstamo, monto de la retención del préstamo que se genera automáticamente de acuerdo al interés que cobre la asociación y las cuotas en las que vaya a cancelar. Buscar retención de préstamo.

Tabla 3-45 Caso de Uso Generación de Retención de Préstamos.
Fuente: El Autor, 2013

Especificación caso de uso: agregar nueva retención de préstamos.

Caso de uso por mes	Agregar Nueva Retención de Préstamos.
Objetivo	Crear la interfaz agregar nueva retención de préstamos que se le debe descontar al asociado de los préstamos que se le ha otorgado.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> • Se debe ingresar préstamos en el menú de préstamos. • El préstamo debe encontrarse activo. • El número de cuotas debe ser mayor a cero.
Acciones Básicas	<ul style="list-style-type: none"> • Esta opción se utiliza si se requiere ingresar de forma manual la retención de un préstamo.

Tabla 3-46 Caso de Uso Agregar Nueva Retención de Préstamos.

Fuente: El Autor, 2013

Especificación caso de uso: eliminar o actualizar retención de préstamos.

Caso de uso por mes	Eliminar o Actualizar Retención de Préstamos
Objetivo	<ul style="list-style-type: none"> • Eliminar retención de préstamos al asociado del préstamo que se le ha otorgado. • Actualizar retención de préstamos al asociado del préstamo que se le ha otorgado.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> • Se debe generar la retención del préstamo. • Se puede eliminar o actualizar la retención en el plazo de 15 días de haberse realizado la retención.
Acciones Básicas	<ul style="list-style-type: none"> • Buscar la retención de préstamos en la lista de retención de préstamos. • Seleccionar la retención de préstamos que se desea eliminar o actualizar. • Eliminar la retención de préstamos seleccionada. • Actualizar la retención de préstamos seleccionada.

Tabla 3-47 Caso de Uso Eliminar o Actualizar Retención de Préstamos.

Fuente: El Autor, 2013

3.2.1.11 DIAGRAMA DE BUSCAR RETENCIONES DE PRÉSTAMOS.

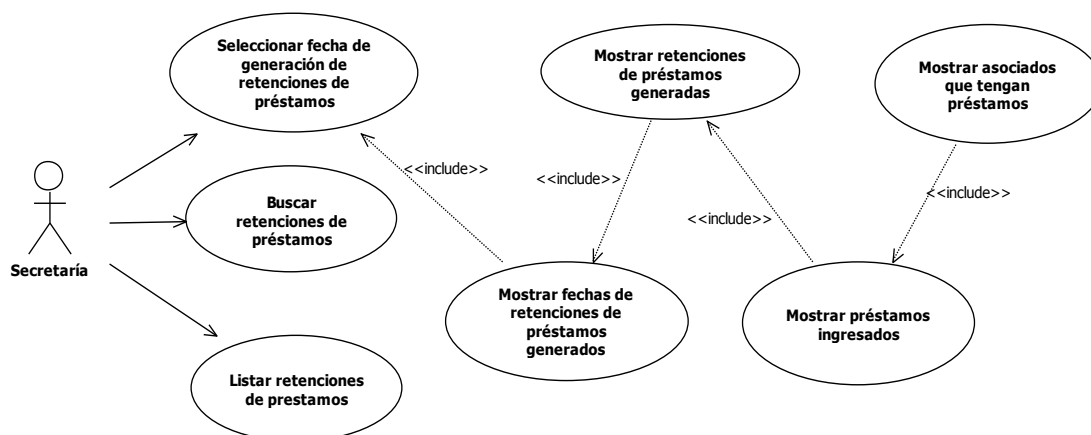


Figura 3-11 Diagrama Buscar Retenciones de Préstamos.

Fuente: El Autor, 2013

Especificación caso de uso: buscar retenciones de préstamos.

Caso de uso por mes	Buscar retenciones de préstamos.
Objetivo	Crear la interfaz de Buscar retenciones de préstamos que se han generado en una fecha determinada.
Actores	Secretaría
Precondiciones	<ul style="list-style-type: none"> Se debe tener retenciones de préstamos generadas.
Acciones Básicas	<ul style="list-style-type: none"> Seleccionar una fecha de generación de retenciones de préstamos. Buscar las retenciones de préstamos en la lista de retenciones de préstamos.

Tabla 3-48 Caso de Uso Buscar Retenciones de Préstamos.

Fuente: El Autor, 2013

3.2.1.12 DIAGRAMA DE GENERACIÓN DE RETENCIÓN DE APORTES.

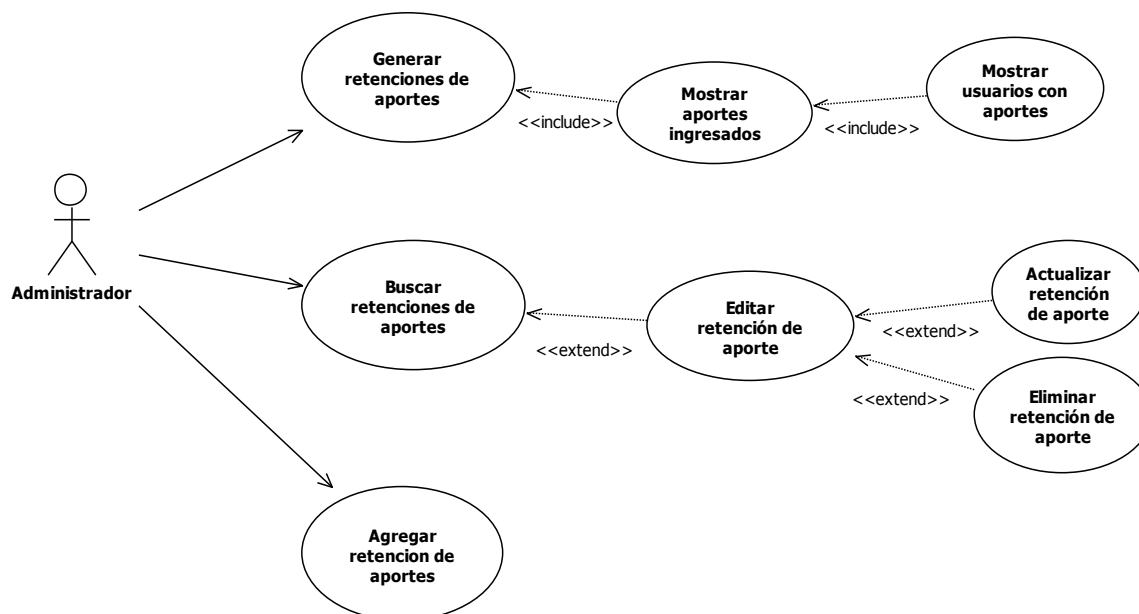


Figura 3-12 Diagrama Generación de Retención de Aportes.

Fuente: El Autor, 2013

Especificación caso de uso: generación de retención de aportes para el mes en curso.

Caso de uso por mes	Generación de Retención de Aportes para el mes en curso.
Objetivo	Crear la interfaz de Generación de Retención de Aportes que se les descuenta a los empleados de la presidencia de la República del Ecuador por pertenecer a la asociación y los servicios que tiene la asociación.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> Se debe ingresar aportes en el menú de aportes. El aporte debe encontrarse activo. El valor del aporte depende si el asociado tiene nombramiento o no lo tiene.
Acciones Básicas	<ul style="list-style-type: none"> Generar retenciones de aportes. La generación de retenciones de aportes se realiza automáticamente. El resultado de esta generación es una lista retenciones de aportes que se detallan con el código de la retención del aportes, código del usuario, código del préstamo, monto de la retención del aporte que se les asigna y se les descuenta automáticamente a todos los asociados.

	<ul style="list-style-type: none"> • Buscar retención de aporte.
--	---

Tabla 3-49 Caso de Uso Generación de Retención de Aportes.
Fuente: El Autor, 2013

Especificación caso de uso: agregar nueva retención de aportes.

Caso de uso por mes	Agregar nueva retención de aportes para el mes en curso.
Objetivo	Crear la interfaz de Agregar nueva retención de aportes para el mes en curso que se generan cada mes y se le debe descontar al asociado.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> • Se requiere una retención del aporte. • Debe estar previamente ingresados los datos del nuevo aporte la tabla aporte. • Se ha eliminado la retención del aporte.
Acciones Básicas	<ul style="list-style-type: none"> • Esta opción se utiliza si se requiere ingresar de forma manual la retención de un aporte.

Tabla 3-50 Caso de Uso Agregar Nueva Retención de Aportes.
Fuente: El Autor, 2013

Especificación caso de uso: eliminar retención de aportes para el mes en curso.

Caso de uso por mes	Eliminar retención de aportes para el mes en curso.
Objetivo	Crear la interfaz Eliminar retención de aportes que se ha generado incorrectamente.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> • Se debe generar la retención del aporte. • Se puede eliminar la retención del aporte en el plazo de 15 días de haberse realizado la retención.
Acciones Básicas	<ul style="list-style-type: none"> • Buscar la retención de aporte en la lista de retención de aportes. • Seleccionar la retención de aportes que se desea eliminar. • Eliminar la retención de aporte seleccionada.

Tabla 3-51 Caso de Uso Eliminar Retención de Aportes para el Mes en Curso.
Fuente: El Autor, 2013

3.2.1.13 DIAGRAMA BUSCAR RETENCIONES DE APORTES.

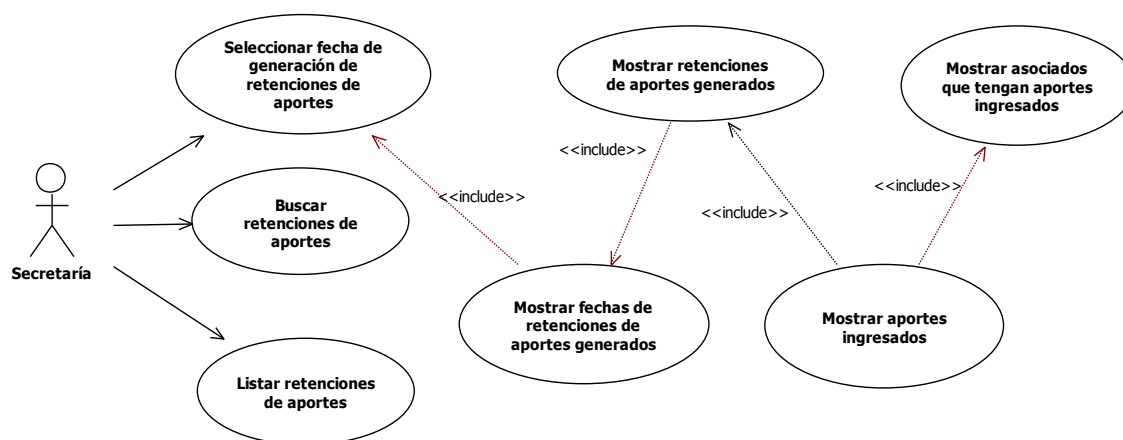


Figura 3-13 Diagrama Buscar Retenciones de Aportes.
Fuente: El Autor, 2013

Especificación caso de uso: buscar retenciones de aportes.

Caso de uso por mes	Buscar retenciones de aportes.
Objetivo	Crear la interfaz Buscar retenciones de aportes que se han generado en una fecha determinada.
Actores	Secretaría
Precondiciones	<ul style="list-style-type: none"> Se debe tener retenciones de aportes generadas.
Acciones Básicas	<ul style="list-style-type: none"> Seleccionar una fecha de generación de retenciones de aportes. Buscar la retención de aporte en la lista de retenciones de aportes.

Tabla 3-52 Caso de Uso Buscar Retenciones de Aportes.
Fuente: El Autor, 2013

3.2.1.14 DIAGRAMA AGREGAR RETENCIÓN DE FACTURA.

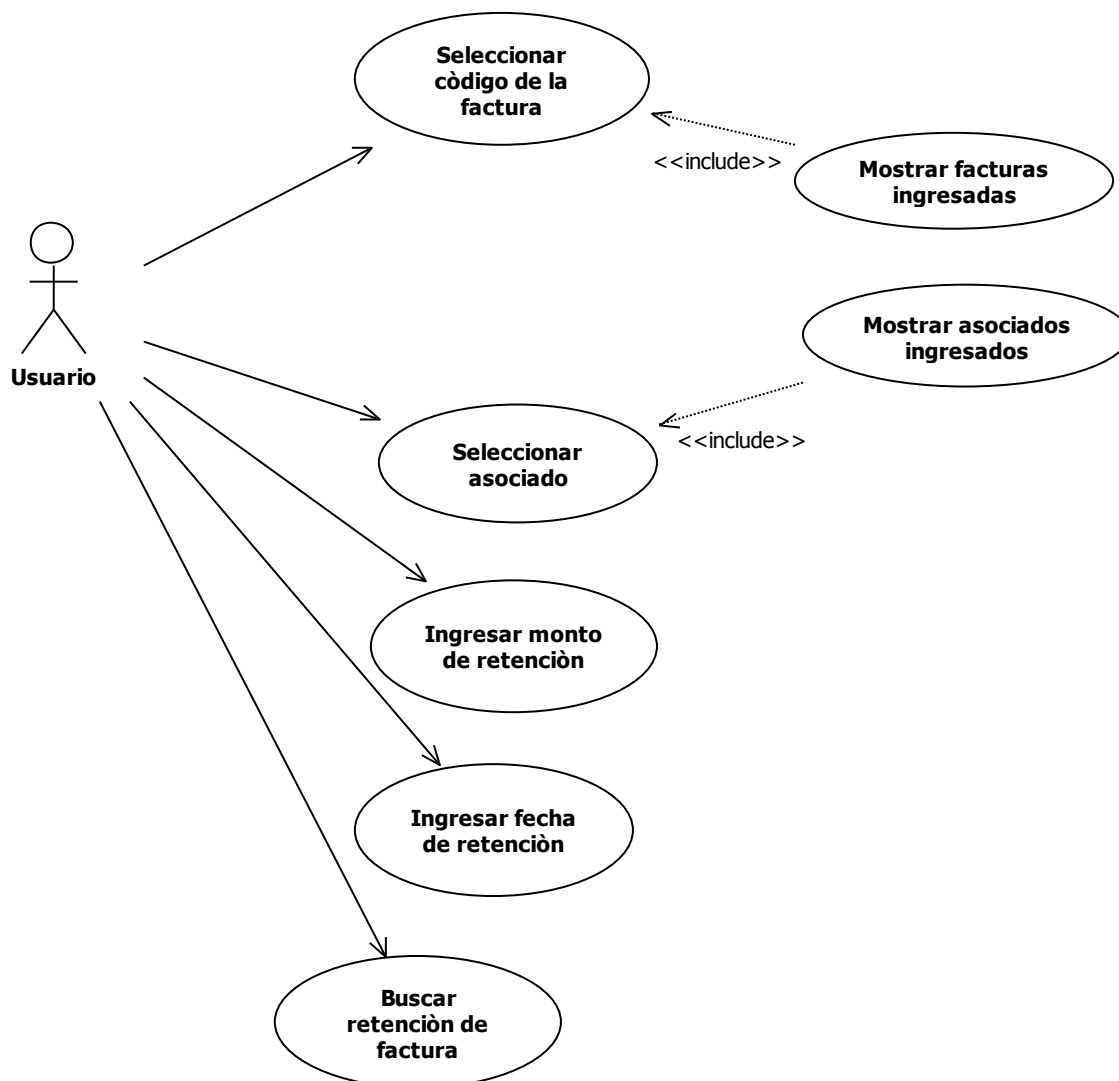


Figura 3-14 Diagrama Agregar Retención de Factura.

Fuente: El Autor, 2013

Especificación caso de uso: agregar retención de factura

Caso de uso por mes	Agregar retención de factura.
Objetivo	Crear la interfaz de Agregar Retención de Factura que los asociados han realizado en cada una de las casas comerciales.
Actores	Usuario.
Precondiciones	<ul style="list-style-type: none"> El administrador debe ingresar al sistema los cheques con los cuales se le va a cancelar la factura.

	<ul style="list-style-type: none"> La secretaria previamente debe ingresar la casa comercial que va a ser facturada.
Acciones Básicas	<ul style="list-style-type: none"> Seleccionar código de factura de la casa comercial. Seleccionar al asociado. Ingresar monto de retención de la factura. Ingresar fecha de la retención de la factura. Buscar la retención de factura en la lista de retenciones de factura.

Tabla 3-53 Caso de Uso Agregar Retención de Factura.

Fuente: El Autor, 2013

3.2.1.15 IAGRAMA DE GENERACIÓN DE RETENCIONES TOTALES.

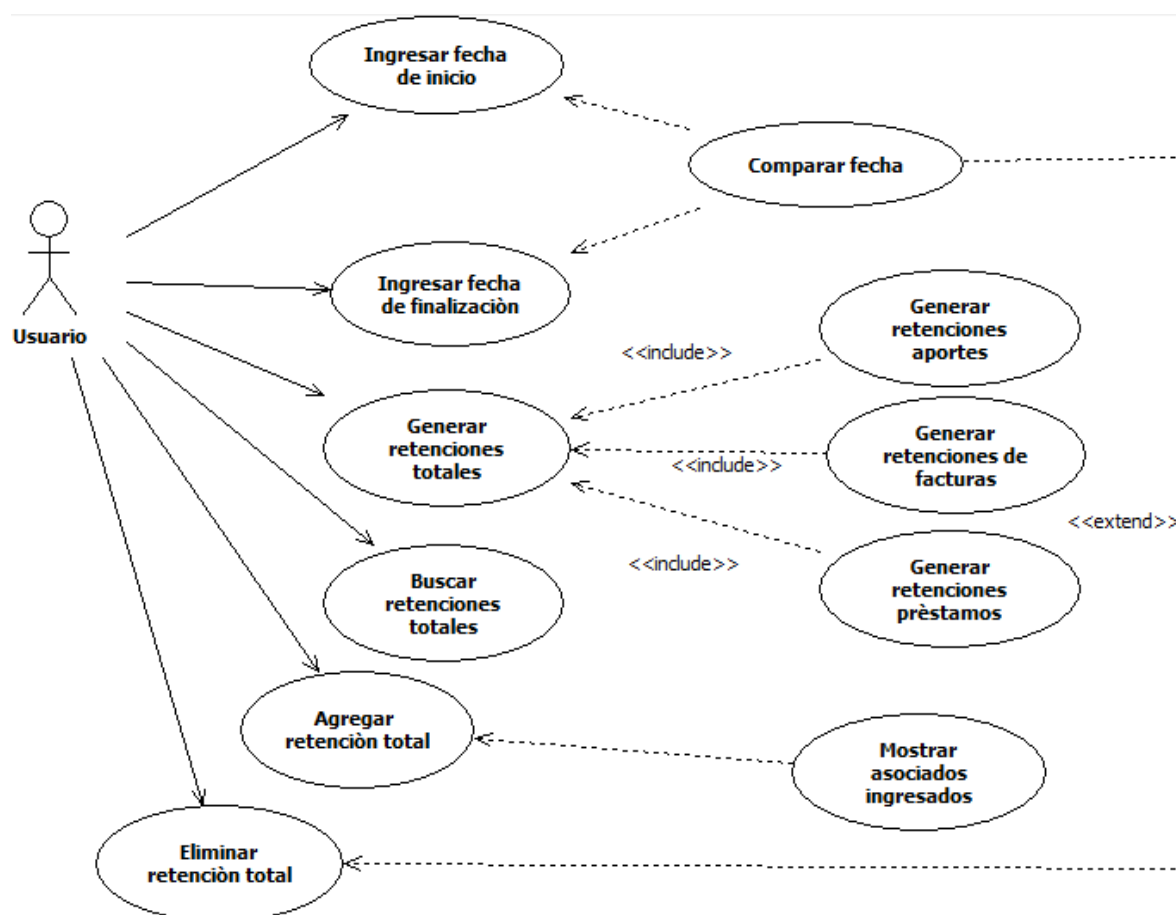


Figura 3-15 Diagrama Generación de Retenciones Totales.

Fuente: El Autor, 2013

Especificación caso de uso: generación de retenciones totales.

Caso de uso por mes	Generación de Retenciones Totales.
Objetivo	<ul style="list-style-type: none"> Crear la interfaz de Generación de Retenciones Totales. Agrupar por mes y por asociado las retenciones de

	<p>préstamos, aporte y factura.</p> <ul style="list-style-type: none"> • Generar un valor total que se le retendrá al asociado.
Actores	Usuario
Precondiciones	<ul style="list-style-type: none"> • Se debe ingresar aportes, préstamos y facturas. • Los aportes, préstamos y facturas deben encontrarse activos.
Acciones Básicas	<ul style="list-style-type: none"> • Ingresar fecha de inicio. • Ingresar fecha de finalización. • Generar retenciones totales. • La generación de retenciones totales se realiza automáticamente. • El resultado de esta generación es una lista que detalla agrupación por mes y por asociado de las retenciones de aportes, facturas y préstamos. • Genera un valor total de retención que se le descontará al asociado cada mes.

Tabla 3-54 Caso de Uso Generación de Retenciones Totales.
Fuente: El Autor, 2013

Especificación caso de uso: agregar retención total.

Caso de uso por mes	Agregar retención total.
Objetivo	Crear la interfaz de Agregar retención total que se le retendrá al usuario cada mes.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> • La retención total no fue generado correctamente. • Debe estar previamente generado las retenciones de aportes, préstamos y facturas. • Se ha eliminado la retención total.
Acciones Básicas	<ul style="list-style-type: none"> • Esta opción se utiliza si se requiere ingresar de forma manual la retención total.

Tabla 3-55 Caso de Uso Agregar Retención Total.
Fuente: El Autor, 2013

Especificación caso de uso: eliminar retención total.

Caso de uso por	Eliminar Retención Total.
------------------------	---------------------------

mes	
Objetivo	Eliminar retención total.
Actores	Administrador
Precondiciones	<ul style="list-style-type: none"> Se debe generar o agregar la retención total. Se puede eliminar la retención en el plazo de 15 días de haberse realizado la retención.
Acciones Básicas	<ul style="list-style-type: none"> Buscar la retención total en la lista de retenciones totales. Seleccionar la fecha de la retención total que se desea eliminar. Eliminar la retención total seleccionada.

Tabla 3-56 Caso de Uso Eliminar Retención Total.
Fuente: El Autor, 2013

3.2.2 DIAGRAMAS DE ESTADO

A continuación se detallan los diagramas de estado que muestran los diferentes estados de un objeto a lo largo del sistema, y los estímulos que provocan los cambios de estado en un objeto.

3.2.2.1 DIAGRAMA DE ESTADO: INICIO DE SESIÓN.

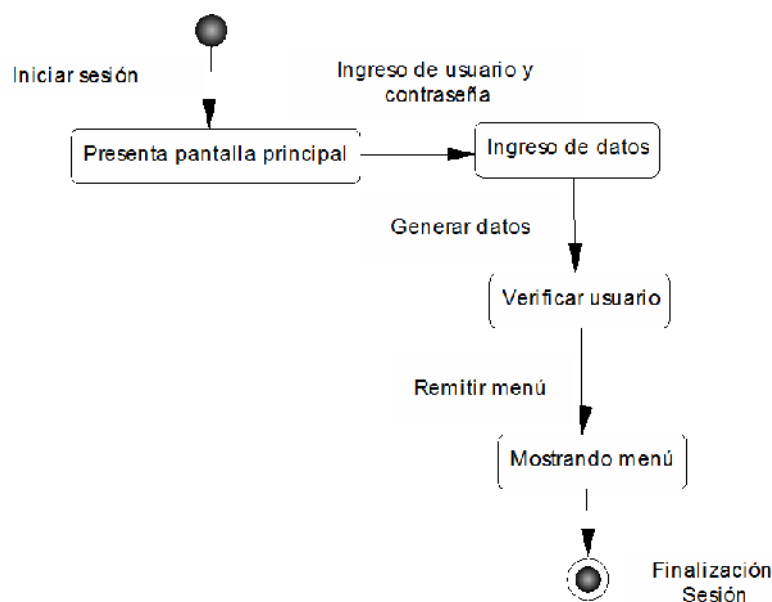


Figura 3-16 Diagrama de Inicio de Sesión.
Fuente: El Autor, 2013

3.2.2.2 DIAGRAMA DE ESTADO: REGISTRO DE FACTURAS.

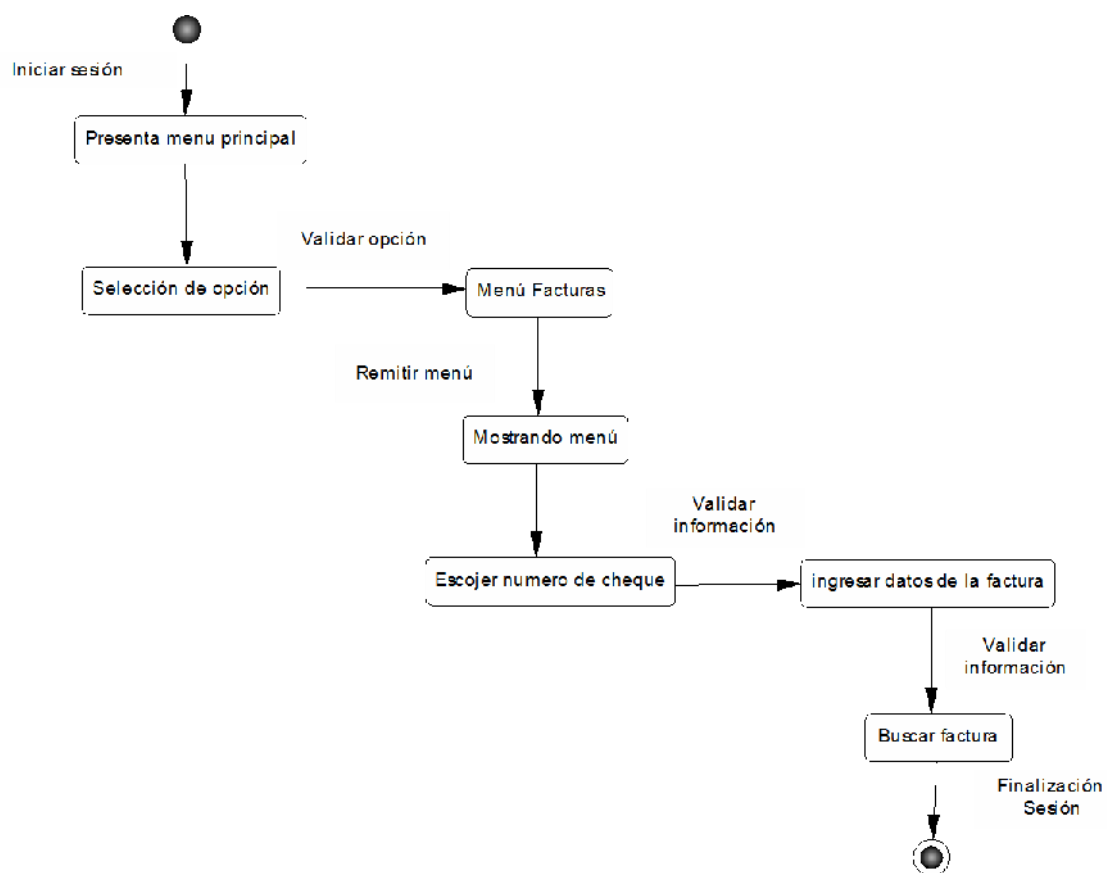


Figura 3-17 Diagrama Registro de Facturas.
Fuente: El Autor, 2013

3.2.2.3 DIAGRAMA DE ESTADO: REGISTRO CUENTA BANCARIA.

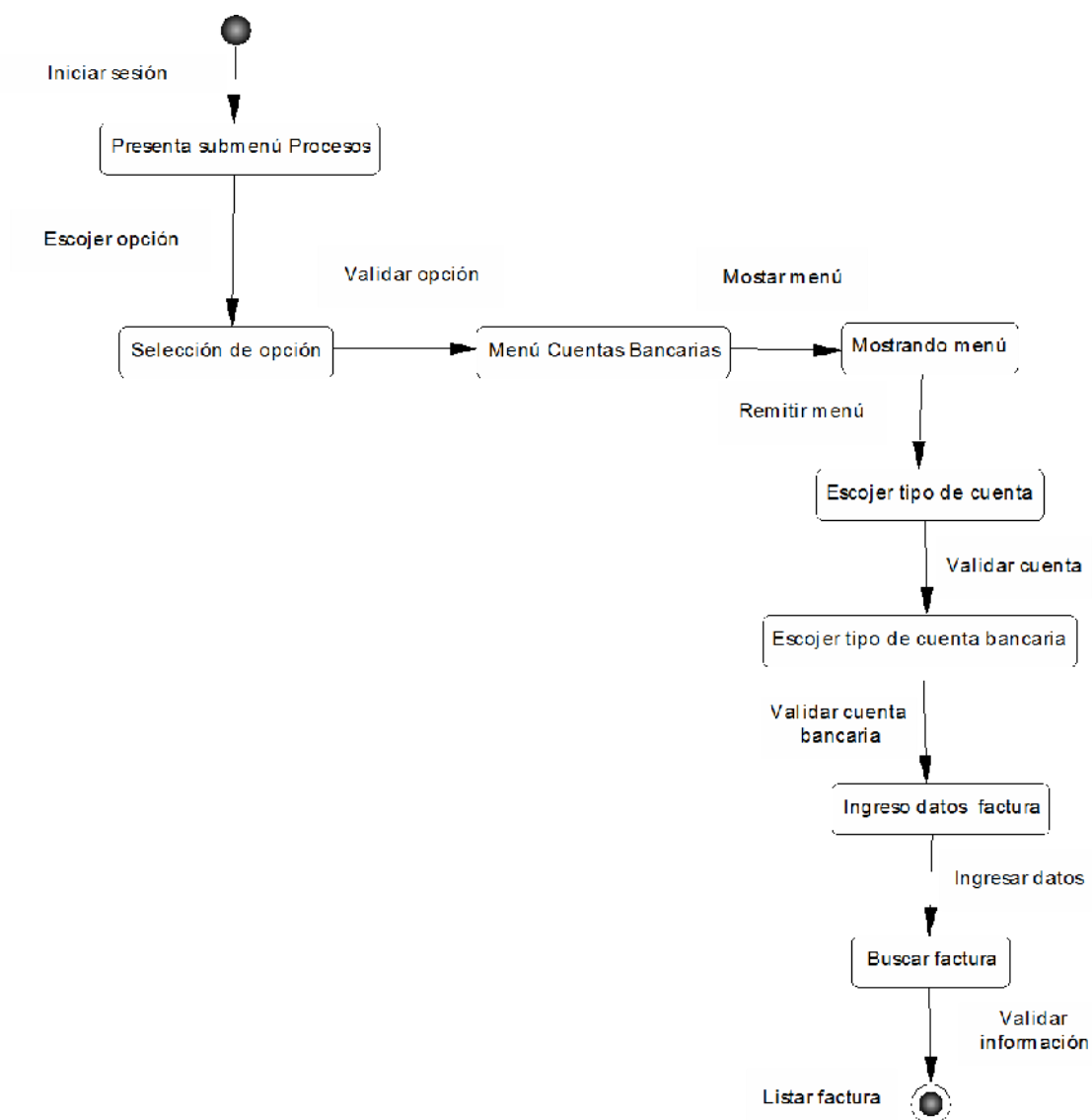


Figura 3-18 Diagrama Registro Cuenta Bancaria.
Fuente: El Autor, 2013

3.2.2.4 DIAGRAMA DE ESTADO: GENERAR RETENCIÓN APORTES.

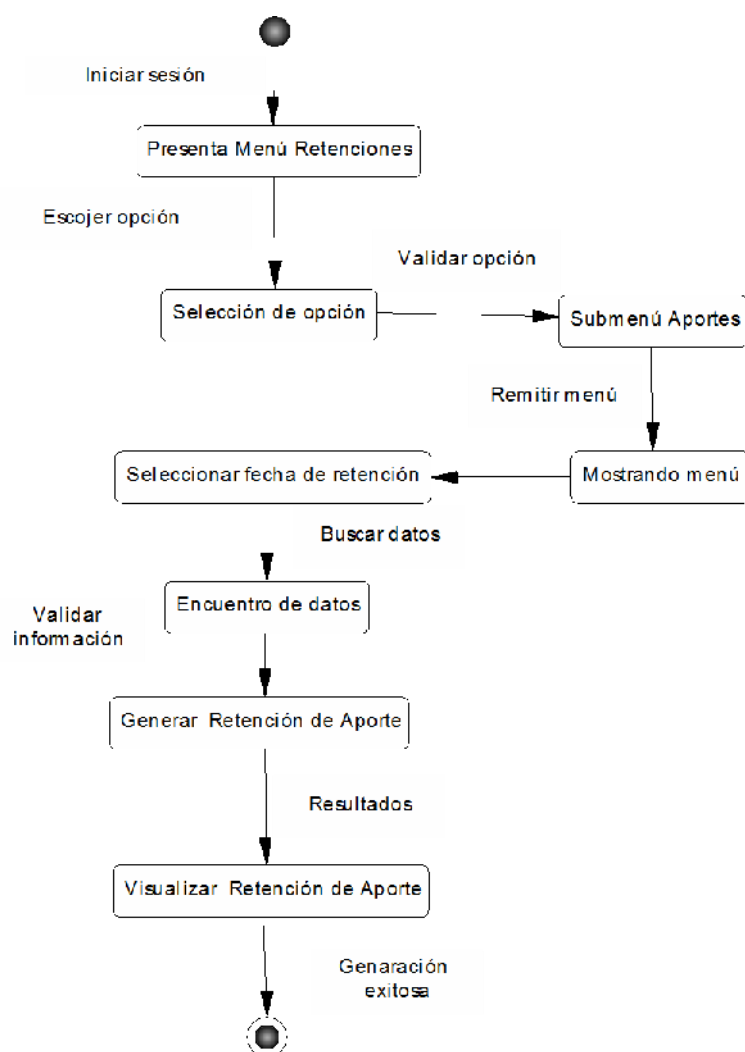


Figura 3-19 Diagrama Generar Retención Aportes.
Fuente: El Autor, 2013

3.2.2.5 DIAGRAMA DE ESTADO: AGREGAR RETENCIÓN FACTURAS.

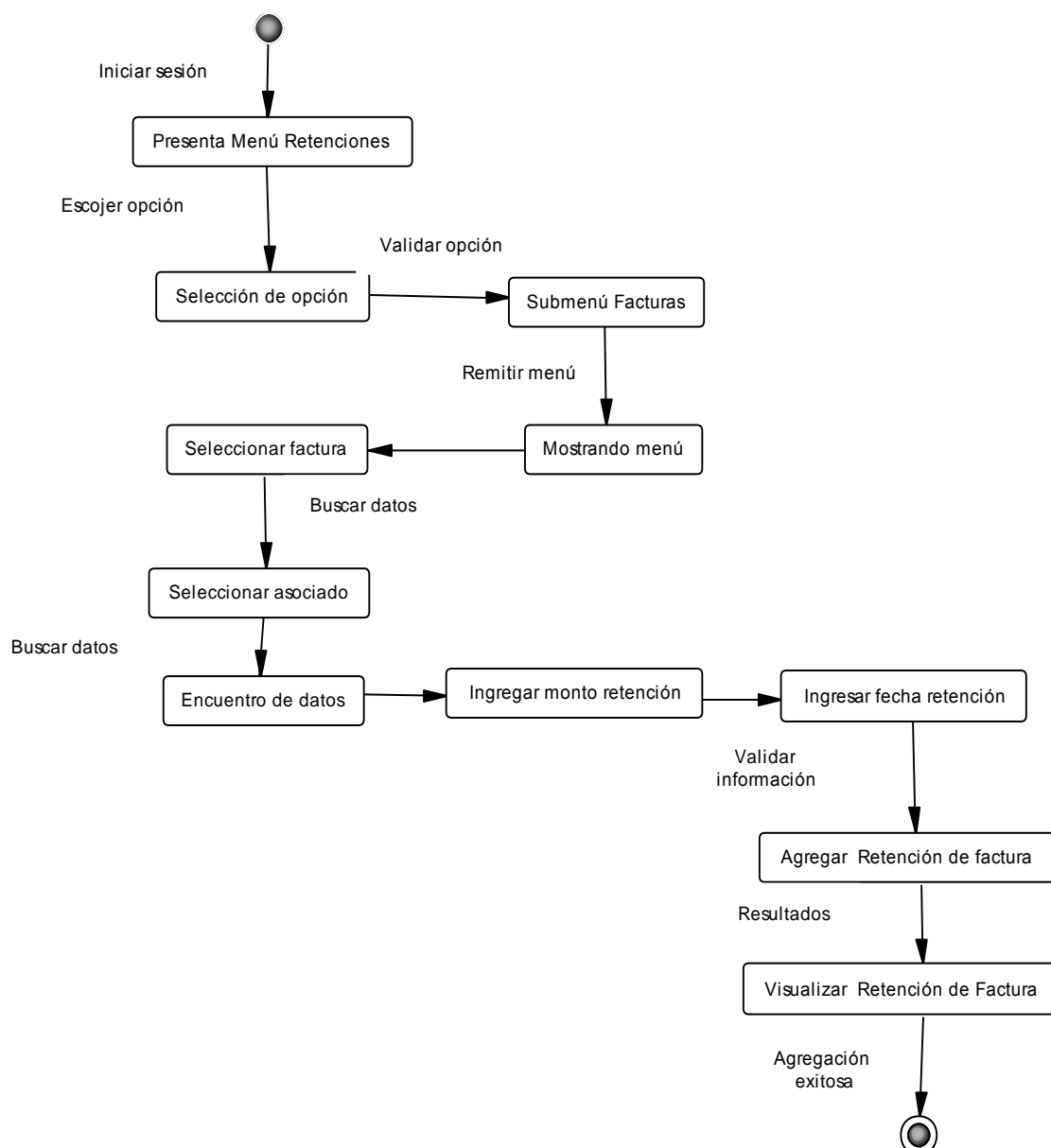


Figura 3-20 Agregar Retención Facturas.
Fuente: El Autor, 2013

3.2.2.6 DIAGRAMA DE ESTADO: GENERAR RETENCIÓN PRÉSTAMOS.

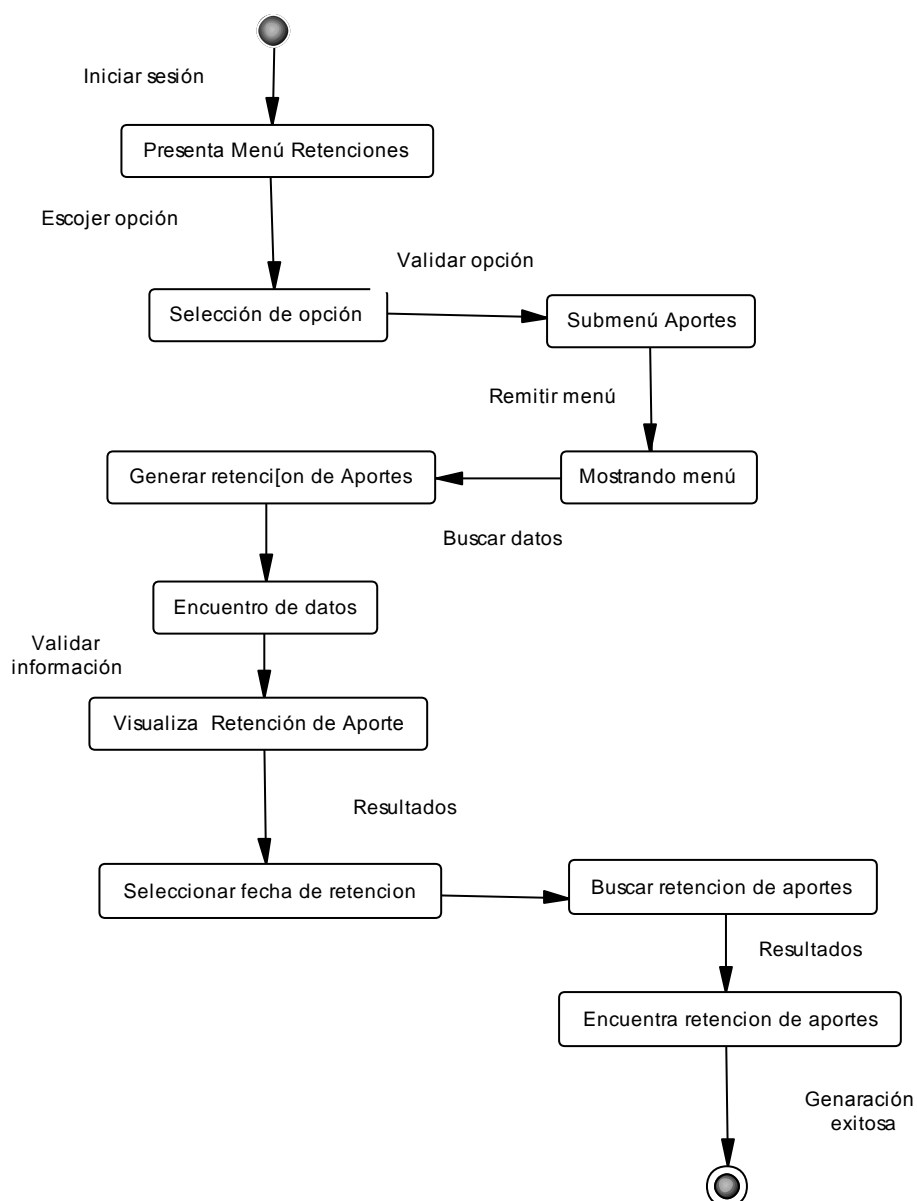


Figura 3-21 Diagrama Generar Retención Préstamos.

Fuente: El Autor, 2013

3.2.2.7 DIAGRAMA DE ESTADO: GENERACIÓN RETENCIÓN TOTAL.

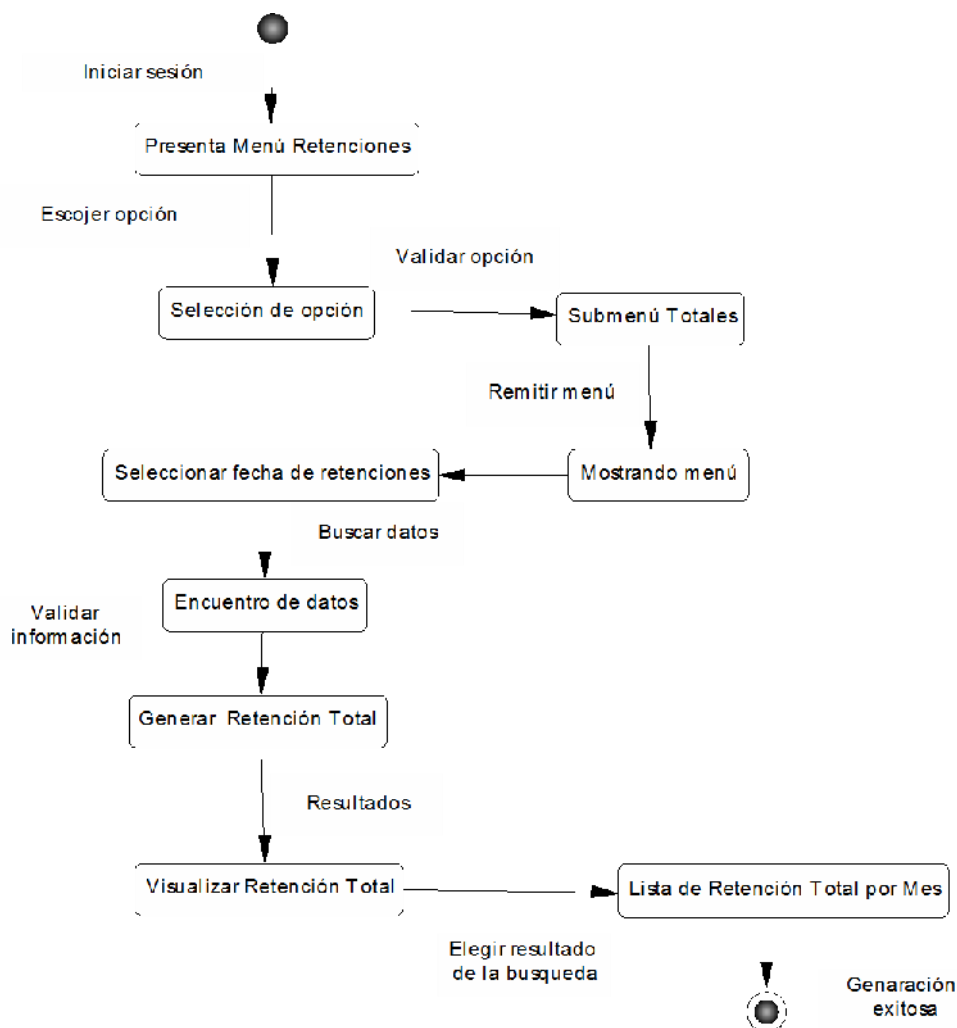


Figura 3-22 Diagrama Generación Retención Total.
Fuente: El Autor, 2013

3.2.3 DIAGRAMAS DE SECUENCIA

Los diagramas de secuencia describen un diseño dinámico del sistema resaltando la ordenación temporal de los mensajes describiendo así cada uno de los escenarios que tiene el sistema.

DIAGRAMA DE SECUENCIA: INICIO DE SESIÓN

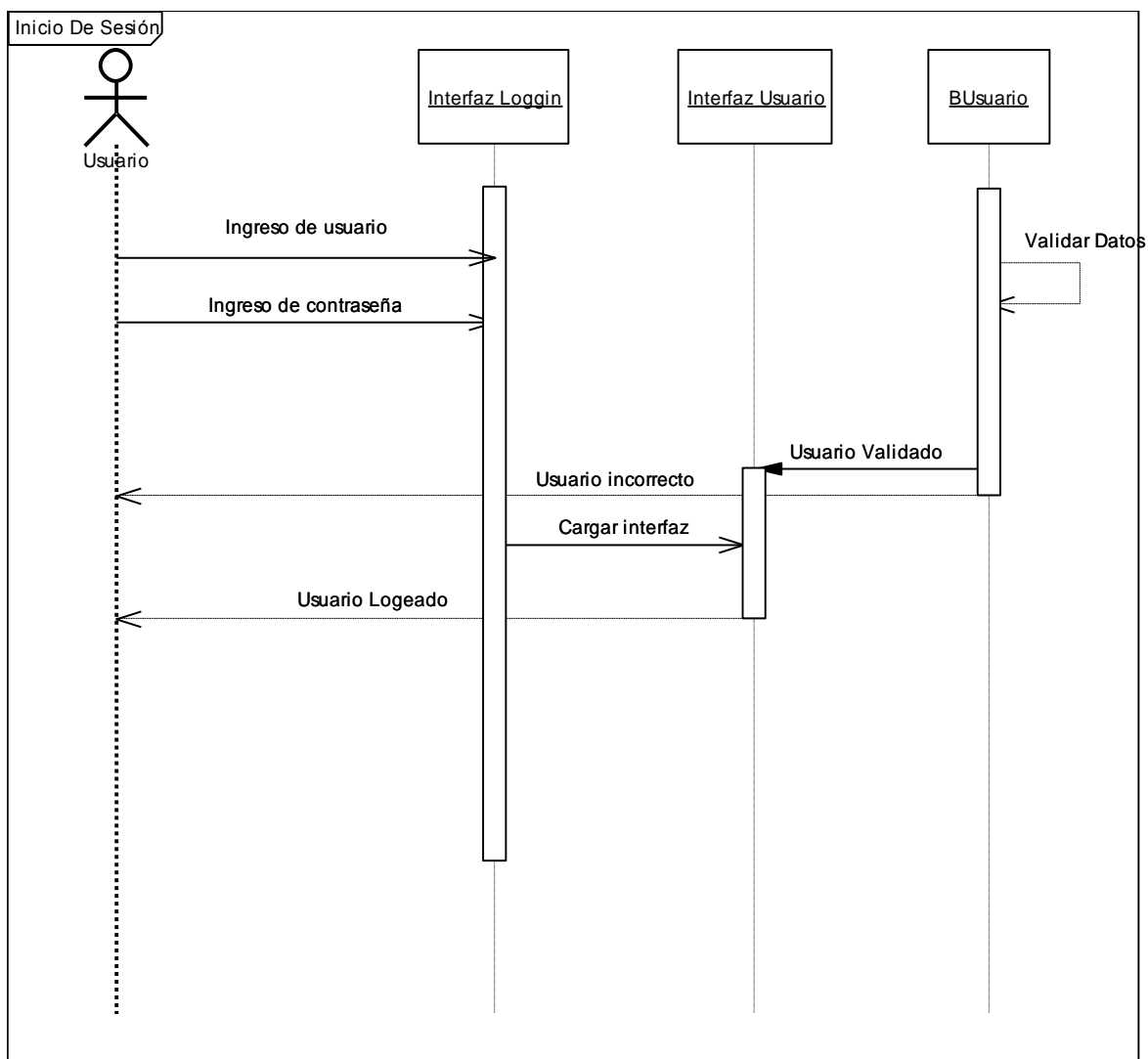


Figura 3-23 Diagrama Generación Inicio de Sesión.
Fuente: El Autor, 2013

DescripciónDiagrama De Secuencia: Inicio De Sesión

Para el inicio de sesión el asociado debe ingresar el número de cédula con su respectiva contraseña, estos datos se envían a la interfaz de usuario para ser validados, si es correcto se despliega un mensaje de éxito y se despliega la interfaz del usuario, caso contrario se despliega un mensaje de error.

DIAGRAMA DE SECUENCIA: REGISTRO DE FACTURAS

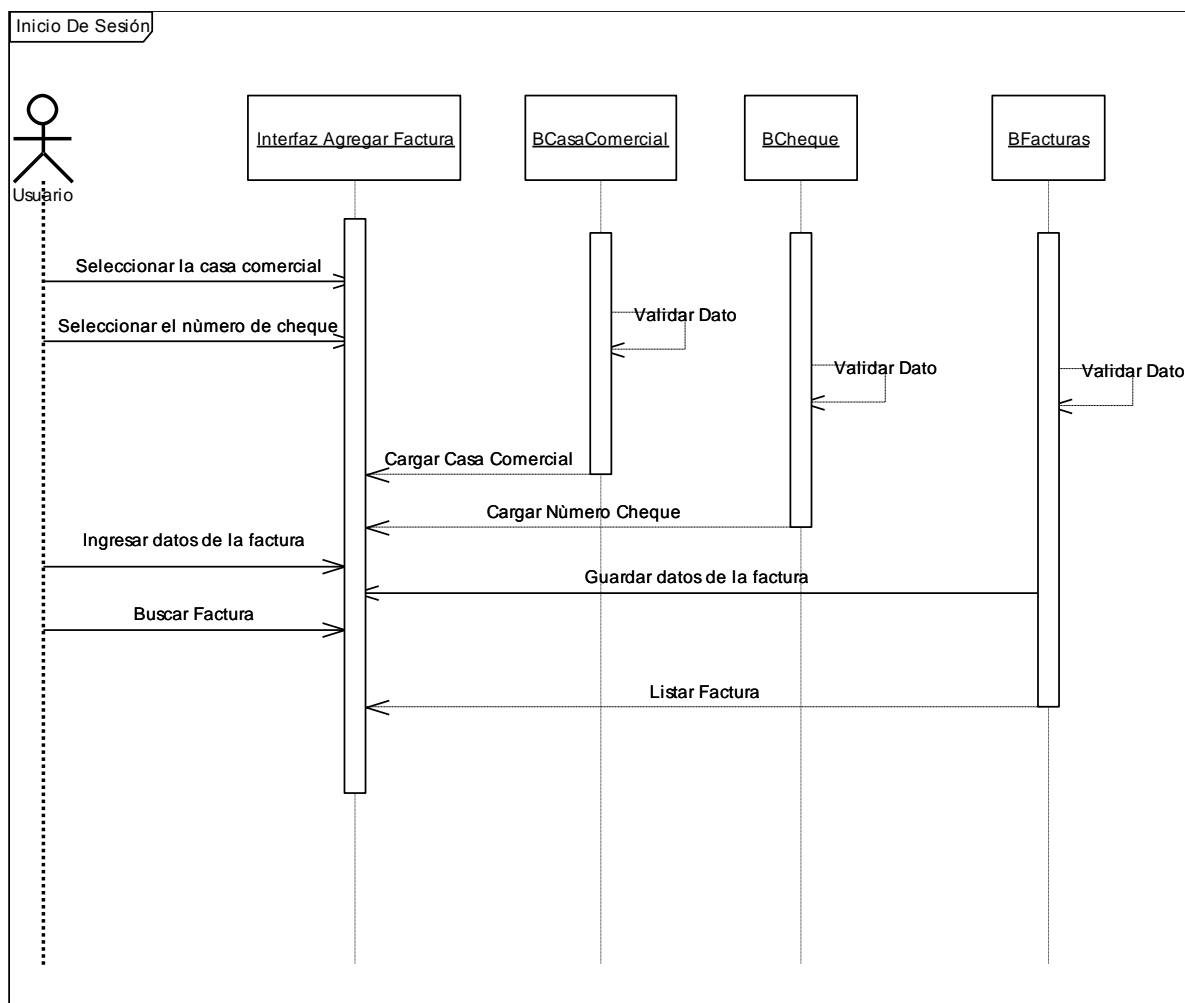


Figura 3-24 Diagrama Generación Registro de Facturas.
Fuente: El Autor, 2013

Descripción Diagrama de Secuencia: Registro de Facturas

El proceso de registro de facturas se inicia con la selección de casa comercial, número de cheque y datos generales de la factura. Dichos datos se envía a la interfaz de casa comercial, cheque y factura para ser validados todos los campos necesarios. Si todo está correcto, se muestra un mensaje de éxito, luego se busca la factura ingresada y se despliega la lista de facturas, caso contrario se despliega un mensaje de error.

DIAGRAMA DE SECUENCIA: REGISTRO DE CUENTAS BANCARIAS

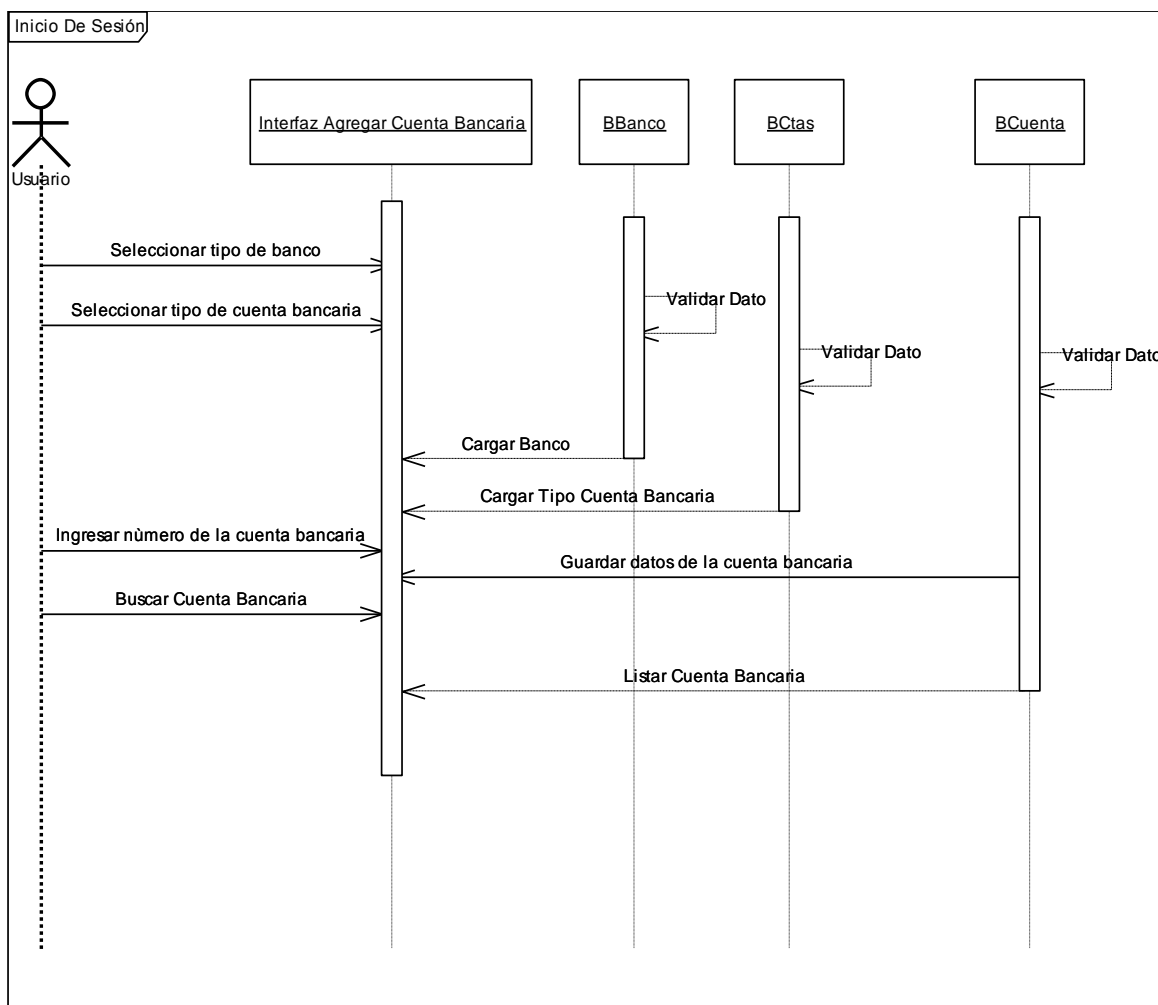


Figura 3-25 Diagrama Registro de Cuentas Bancarias.
Fuente: El Autor, 2013

Descripción Diagrama de Secuencia: Registro de Cuentas Bancarias

Para registrar las cuentas bancarias se selecciona el tipo de banco, tipo de cuenta bancaria, posteriormente se ingresa el número de la cuenta bancaria. Estos datos se envían a la interfaz de banco y de cuentas para que sean validados. Si todo está correcto se despliega un mensaje de éxito de ingreso del registro cuenta bancaria, luego se busca la cuenta bancaria y se despliega una lista de cuentas, caso contrario se despliega un mensaje de error.

DIAGRAMA DE SECUENCIA: GENERACIÓN TOTAL DE RETENCIÓN POR MES

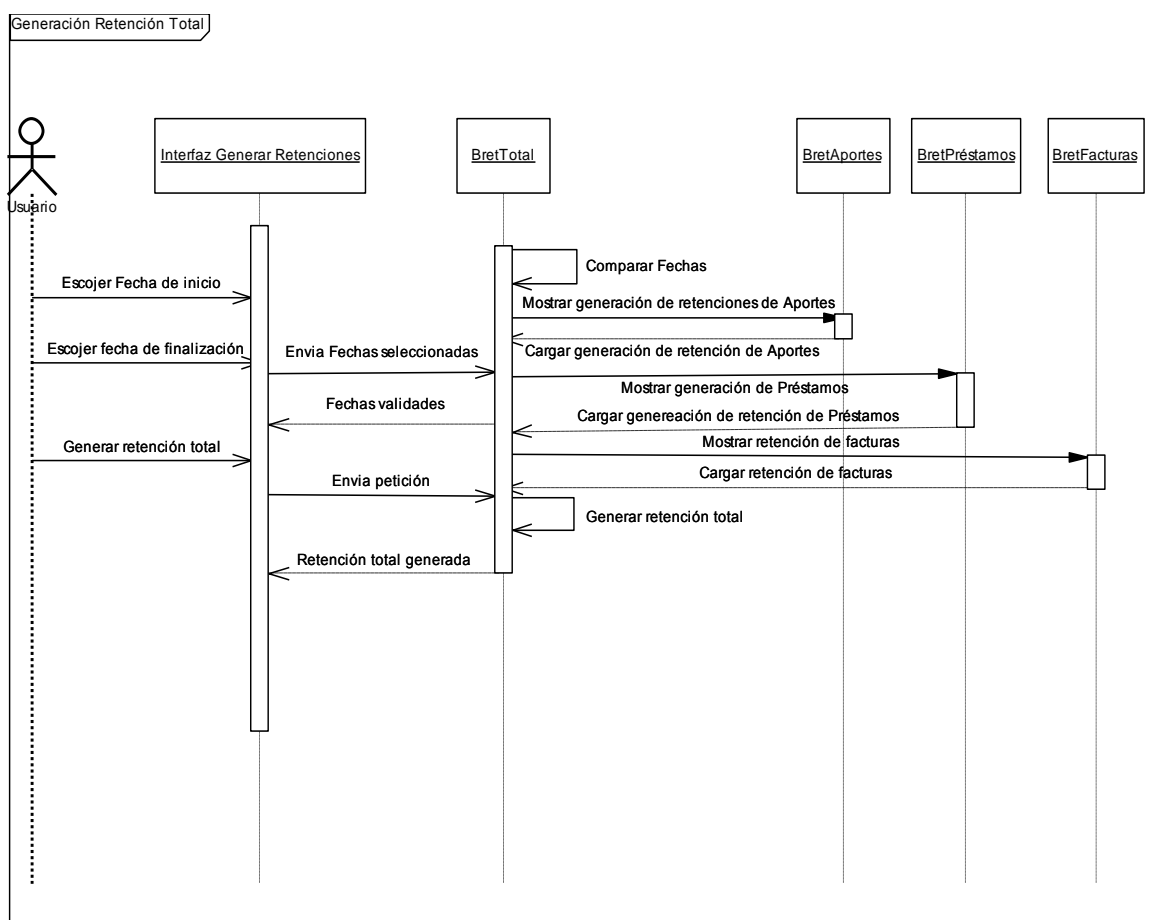


Figura 3-26 Diagrama Generación Retención Total por Mes.
Fuente: El Autor, 2013

Descripción Diagrama de Secuencia: Generación Total de Retención por Mes

La generación retención total por mes se inicia con la selección de fecha de inicio y fecha de finalización. Estos datos se envían a la interfaz de retención de aportes, préstamos y facturas para ser buscadas de acuerdo a las fechas establecidas. Si las fechas son correctas, esta información es enviada a la interfaz de retención total para generar automáticamente la sumatoria una retención general total por cada asociado. En el caso que la validación de las fechas sea incorrecta se despliega un mensaje de error.

3.2.4 DIAGRAMA NAVEGACIONAL

3.2.4.1 ADMINISTRADOR

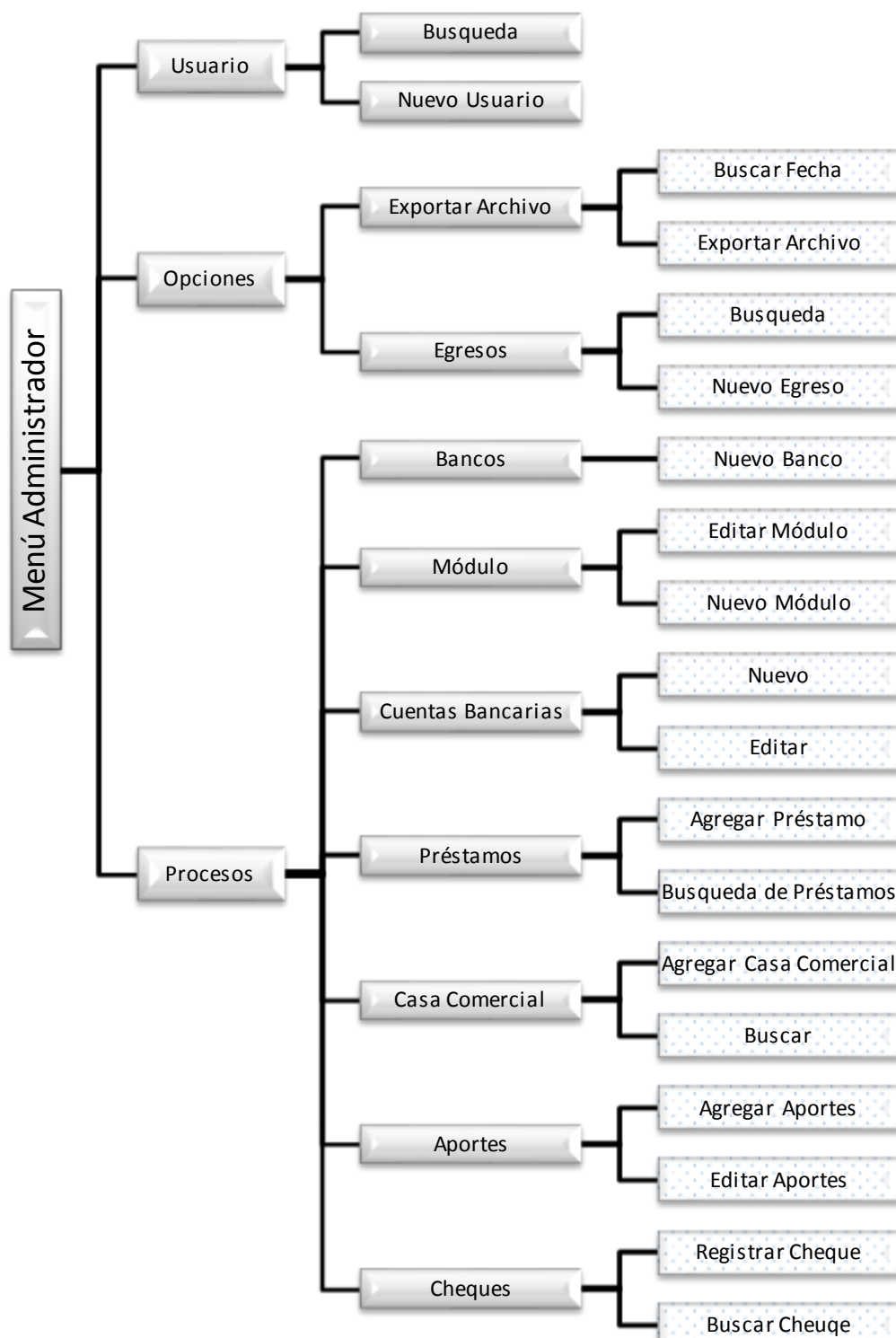


Figura 3-27 Diagrama Navegacional Administrador_1

Fuente: El Autor, 2013

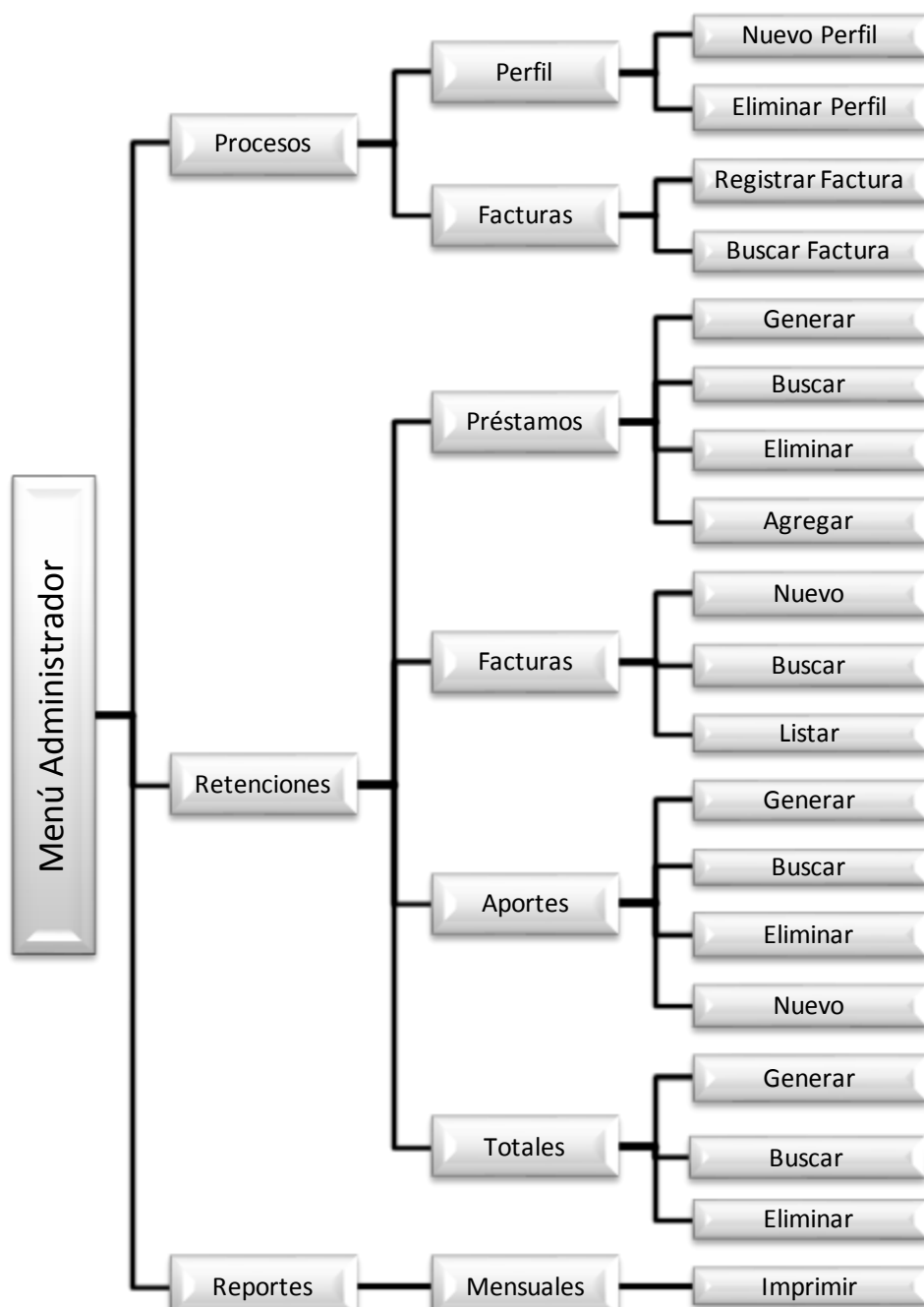


Figura 3-28 Diagrama Navegacional Administrador_2
Fuente: El Autor, 2013

3.2.4.2 SECRETARIA

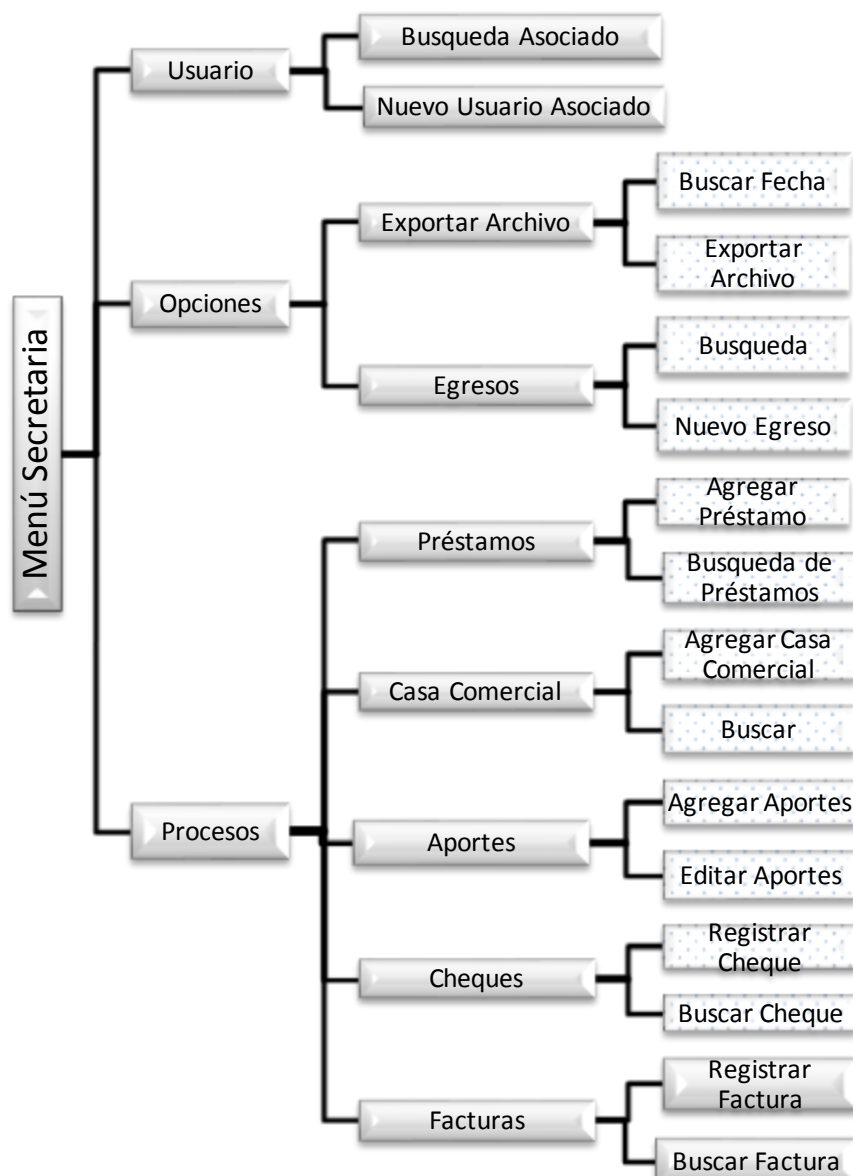


Figura 3-29 Diagrama Navegacional Secretaria_1

Fuente: El Autor, 2013

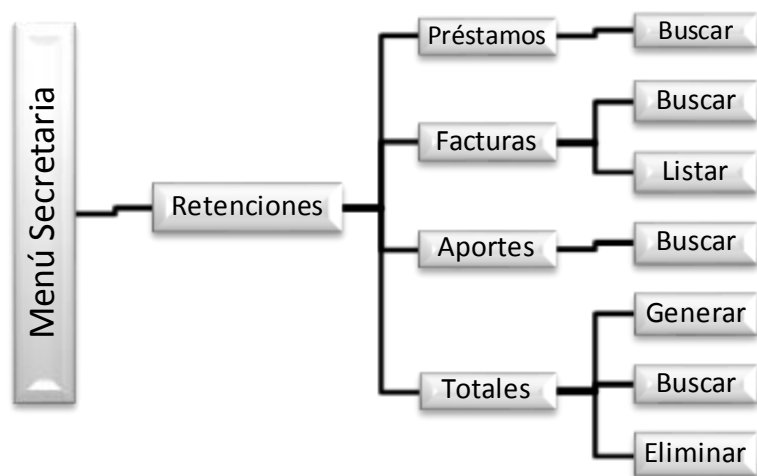


Figura 3-30 Diagrama Navegacional Secretaria_2
Fuente: El Autor, 2013

3.2.4.3 DISPOSITIVO MÓVIL

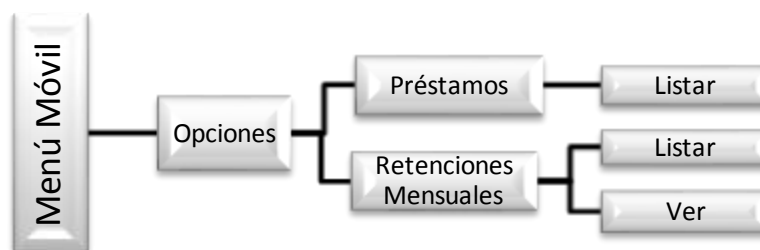


Figura 3-31 Diagrama Navegacional Dispositivo Móvil
Fuente: El Autor, 2013

3.2.4.4 ASOCIADO

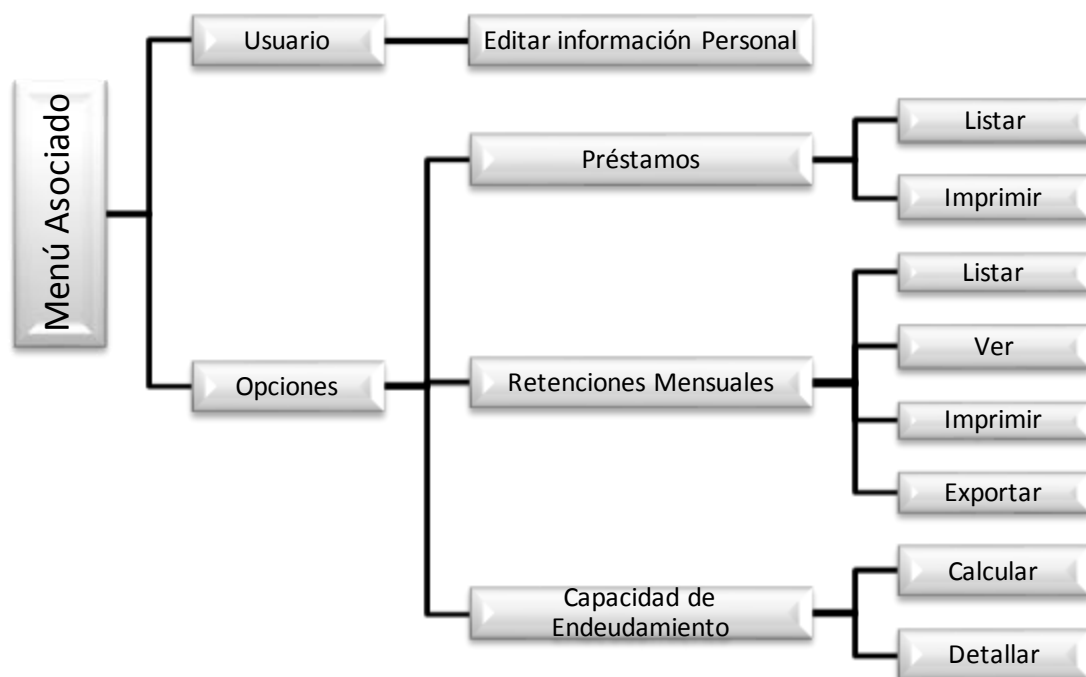


Figura 3-32 Diagrama Navegacional Asociado
Fuente: El Autor, 2013

3.2.5 DIAGRAMA DE CLASES

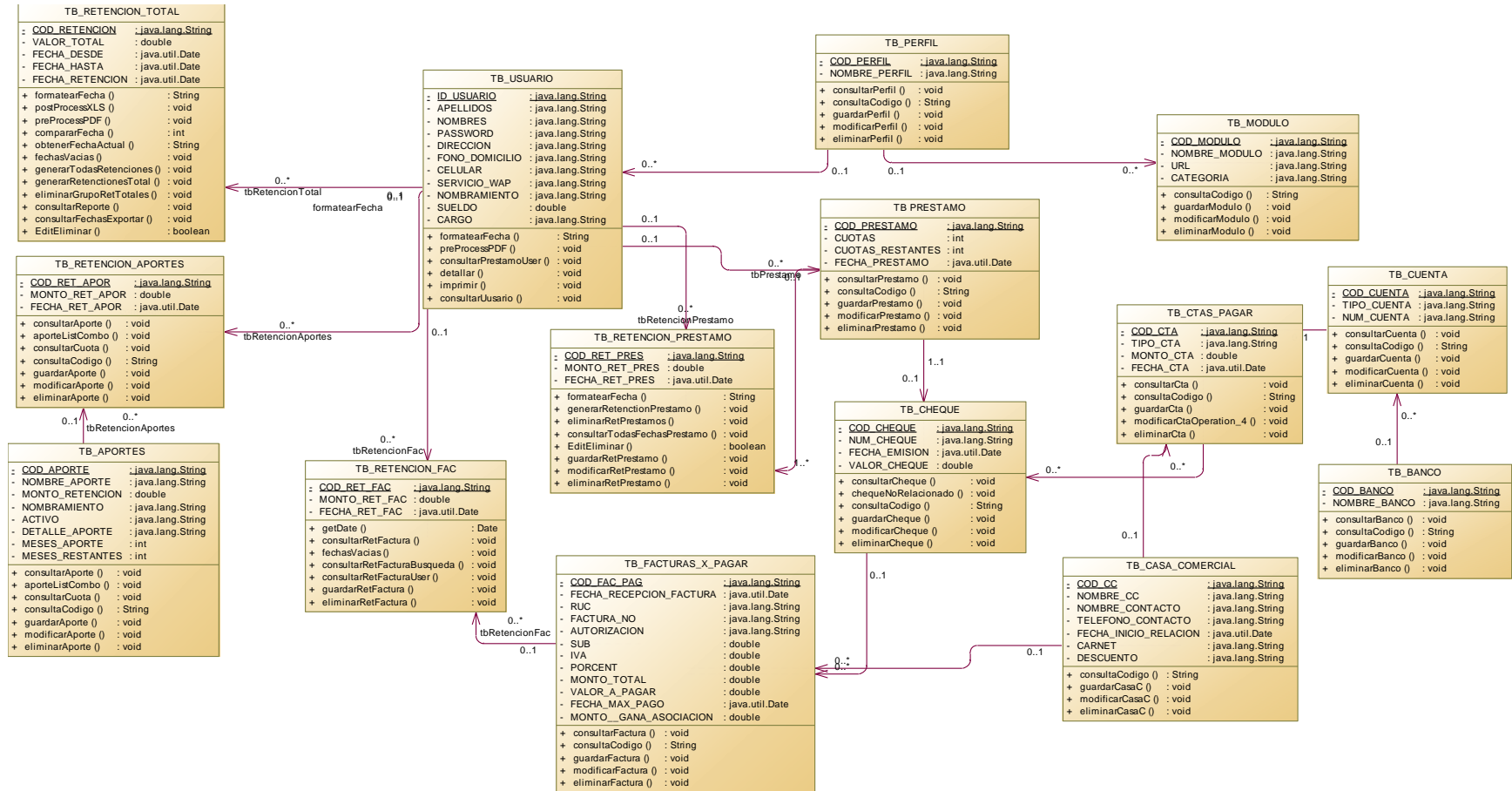


Figura 3-33 Diagrama de Clases
Fuente: El Autor, 2013

3.2.6 DIAGRAMA DE BASE DE DATOS

3.2.6.1 MODELO LÓGICO DE LA BASE DE DATOS

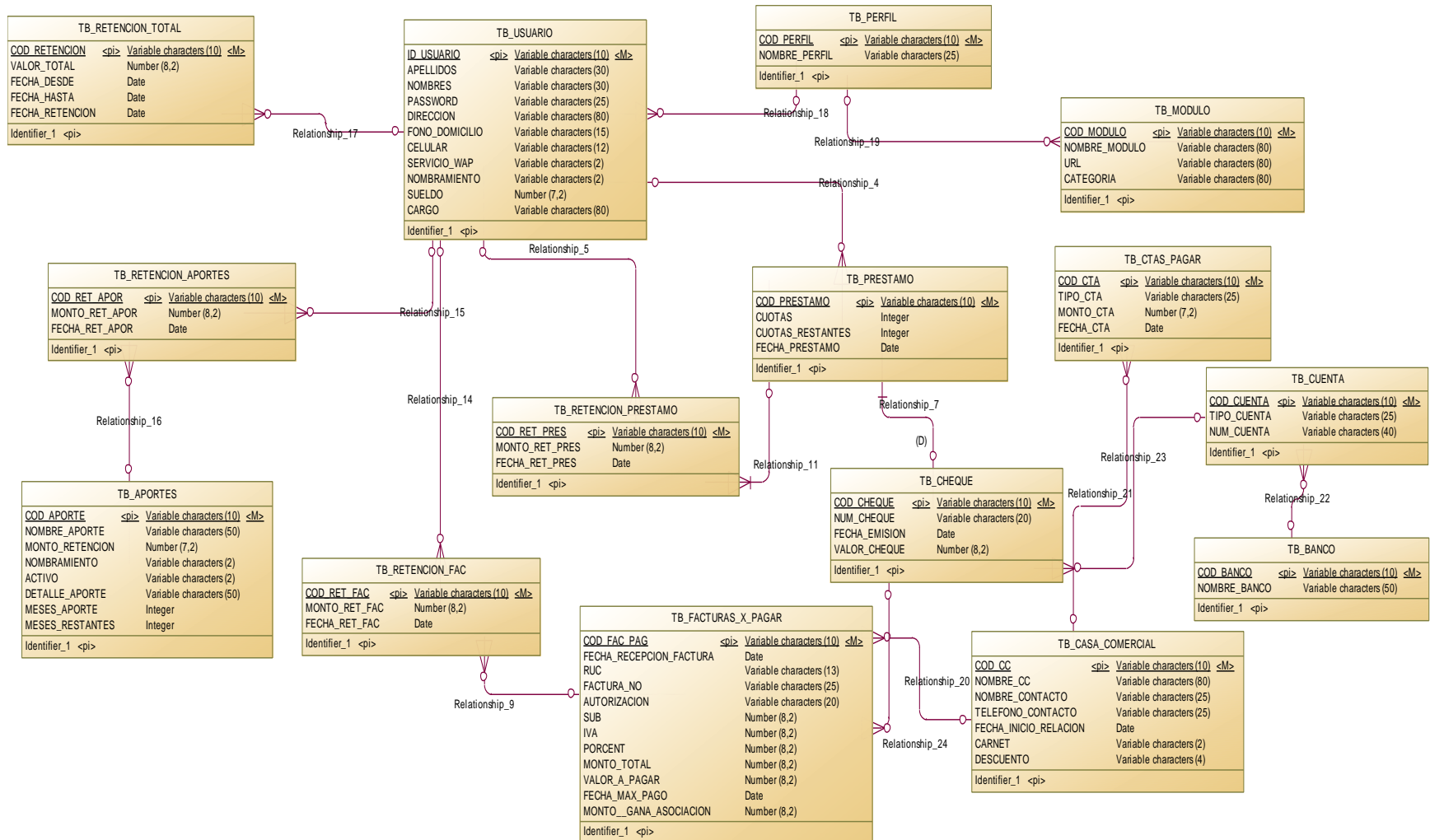


Figura 3-34 Modelo Lógico de Bdd
Fuente: El Autor, 2013

3.2.6.2 MODELO FÍSICO DE LA BASE DE DATOS

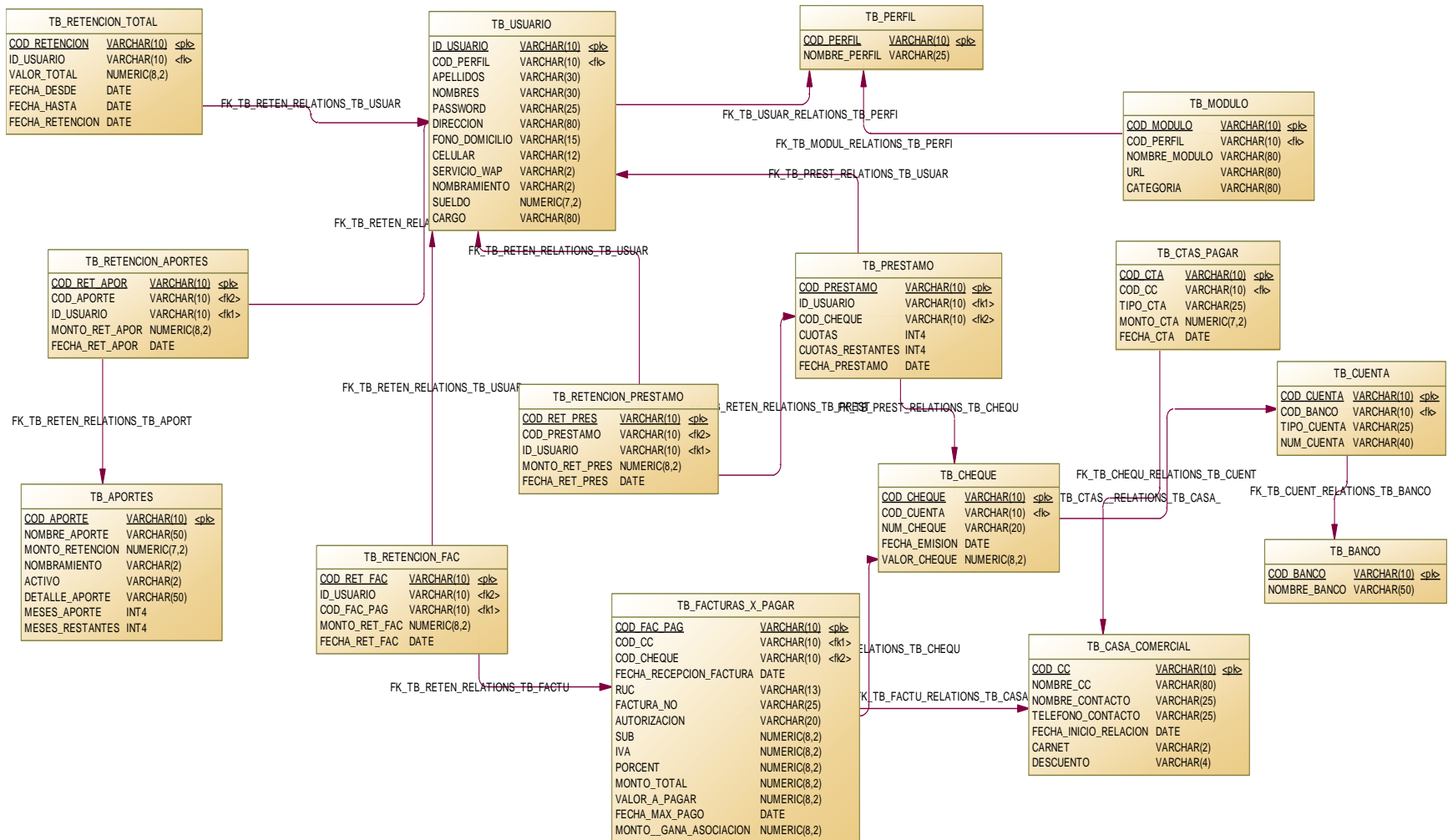


Figura 3-35 Modelo Físico de Bdd
Fuente: El Autor, 2013

3.2.7 DICCIONARIO DE DATOS

El diccionario de datos identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, el cual refiere a la base de datos y sus objetos, describiendo cada una de las tablas que constan en el modelo entidad relación, con cada uno de los campos que contienen mencionadas tablas.

Propósito.

Dar precisión sobre los datos que se manejan en el sistema, evitando así malas interpretaciones o ambigüedades.

Descripción

Detalle	
Nombre	bd_asopresi
Código	bd_asopresi
DBMS	PostgreSQL

Tabla 3-57: Detalle Base de Datos
Fuente: El Autor, 2013

3.2.7.1 DESCRIPCIÓN GENERAL DE LAS TABLAS

Nombre	Descripción
tb_aportes	Detalla los aportes obligatorios de la asociación con un mismo valor mensual a ser descontado
tb_banco	Muestra los bancos con los que se relaciona la asociación.
tb_casa_comercial	Almacena los datos informativos de las casas comerciales.
tb_cheque	Registro de los cheques generados por la asociación.
tb_ctas_pagar	Describe los egresos o gastos que genera la asociación.
tb_cuenta	Detalla las cuentas bancarias que maneja la asociación.
tb_facturas_x_pagar	Almacena las facturas de las casas comerciales.
tb_modulo	Descripción de los módulos desarrollados para el sistema.
tb_perfil	Muestra los perfiles que tiene la asociación.
tb_prestamo	Describe los préstamos emitidos a los asociados.
tb_retencion_aportes	Muestra todas las retenciones de los aportes.
tb_retencion_fac	Almacena las retenciones de las casas comerciales.
tb_retencion_prestamo	Muestra el detalle de las retenciones de los préstamos efectuados.
tb_retencion_total	Muestra la retención total que se le descuenta al asociado,
tb_usuario	Almacena los datos informativos de los asociados de la presidencia.

Tabla 3-58: Descripción General de las Bases de Datos
Fuente: El Autor, 2013

DESCRIPCIÓN COMPLETA DE LAS TABLAS

3.2.7.2 TABLA TB_APORTES

Detalla los aportes obligatorios de la asociación con un mismo valor mensual a ser descontado.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_aporte	charactervarying (10)	Si		Not Null	Identificador único de código de aporte
nombre_aporte	charactervarying (50)		Si	Null	Nombre del aporte.
monto_retencion	numeric(7,2)			Null	Monto de la retención
nombramiento	charactervarying (2)			Null	Nivel jerárquico del asociado
activo	charactervarying (2)			Null	Estado del asociado.
detalle_aporte	charactervarying (50)			Null	Detalle del aporte
meses_aporte	Integer			Null	Valor del aporte en meses
meses_restantes	Integer			Null	Valor restante del aporte en meses

Tabla 3-59: Descripción Tabla Tb_Aportes
Fuente: El Autor, 2013

3.2.7.3 TABLA TB_BANCO

Muestra los bancos con los que se relaciona la asociación.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_banco	charactervarying(10)	Si		Not Null	Identificador único de código de banco
nombre_banco	charactervarying(50)			Null	Nombre del banco

Tabla 3-60: Descripción Tabla Tb_Banco
Fuente: El Autor, 2013

3.2.7.4 TABLA TB_CASA_COMERCIAL

Almacena los datos informativos de las casas comerciales.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_cc	charactervarying(10)	Si		Not Null	Identificador único de la tabla tb_casa_

					comercial
nombre_cc	character varying(50)			Null	Nombre de la casa comercial.
nombre_contacto	Numeric(7,2)			Null	Nombre del contacto de la casa comercial
telefono_contacto	character varying(2)			Null	Teléfono del contacto de la casa comercial
fecha_inicio_relacion	character varying(2)			Null	Fecha de inicio de relación de la casa comercial
Carnet	character varying(50)			Null	Carnet de la casa comercial
descuento	Integer			Null	Descuento de la casa comercial

Tabla 3-61: Descripción Tabla Tb_Casa_Comercial
Fuente: El Autor, 2013

3.2.7.5 TABLA TB_CHEQUE

Registro de los cheques generados por la asociación.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_cheque	character varying(10)	Si		Not Null	Identificador único de la tabla tb_cheque
num_cheque	character varying(20)			Null	Número de cheque
fecha_emision	date			Null	Fecha de emisión de cheque
valor_cheque	numeric(8, 2)			Null	Valor del cheque
cod_cuenta	character varying(10)		Si	Null	Identificador único de la tabla tb_cuenta

Tabla 3-62: Descripción Tabla Tb_Cheque
Fuente: El Autor, 2013

3.2.7.6 TABLA TB_CTAS_PAGAR

Describe los egresos o gastos que genera la asociación.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_cta	character varying(10)	Si		Not Null	Identificador único de la tabla tb_cuenta
cod_cc	character varying(10)		Si	Null	Identificador único de la tabla tb_casa_comercial

tipo_cta	character varying(25)			Null	Tipo de cuenta
monto_cta	Numeric(7, 2)			Null	Monto de la cuenta a pagar
fecha_cta	date			Null	Fecha de la cuenta a pagar

Tabla 3-63: Descripción Tabla Tb_Ctas_Pagar
Fuente: El Autor, 2013

3.2.7.7 TABLA TB_CUENTA

Detalla las cuentas bancarias que maneja la asociación.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_cuenta	character varying(10)	Si		Not Null	Identificador único de la tabla tb_cuenta
cod_banco	character varying(10)		Si	Null	Identificador único de la tabla tb_banco
tipo_cuenta	character varying(20)			Null	Tipo de cuenta
num_cuenta	character varying(40)			Null	Número de cuenta

Tabla 3-64: Descripción Tabla Tb_Cuenta
Fuente: El Autor, 2013

3.2.7.8 TABLA TB_FACTURAS_X_PAGAR

Almacena las facturas de las casas comerciales.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_fac_pag	character varying(10)	Si		Not Null	Identificador único de la tabla tb_facturas_x_pagar
cod_cheque	character varying(10)		Si	Null	Identificador único de la tabla tb_cheque
cod_cc	character varying(10)		Si	Null	Identificador único de la tabla tb_casa_comercial
fecha_recepcion_factura	date			Null	Fecha de la recepción de la factura
ruc	character varying(13)			Null	Registro único del contribuyente
factura_no	character varying(25)			Null	Número de factura
autorización	character varying(15)			Null	autorización de la factura
sub	numeric(8,			Null	Subtotal de la

	2)				factura
iva	numeric(8, 2)			Null	Impuesto al valor agregado
porcent	numeric(8, 2)			Null	Valor de facturas que no cobran iva
monto_total	numeric(8, 2)			Null	Monto Total de la factura
valor_a_pagar	numeric(8, 2)			Null	Valor a pagar de la factura
fecha_max_pago	date			Null	Fecha máxima de pago
monto_gana_asociacion	numeric(8, 2)			Null	Monto que gana la asociación

Tabla 3-65: Descripción Tabla Tb_Facturas_X_Pagar
Fuente: El Autor, 2013

3.2.7.9 TABLA TB_MÓDULO

Descripción de los módulos desarrollados para el sistema.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_modulo	character varying(10)	Si		Not Null	Identificador único la tabla tb_modulo
cod_perfil	character varying(10)		Si	Null	Identificador único de la tabla tb_perfil
nombre_modulo	character varying(80)			Null	Nombre de módulo
url	character varying(80)			Null	Url
categoria	character varying(80)			Null	Categoría

Tabla 3-66: Descripción Tabla Tb_Modulo
Fuente: El Autor, 2013

3.2.7.10 TABLA TB_PERFIL

Muestra los perfiles que tiene la asociación.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_perfil	character varying(10)	Si		Not Null	Identificador único de la tabla tb_perfil
nombre_perfil	character varying(25)			Null	Nombre del perfil

Tabla 3-67: Descripción Tabla Tb_Perfil
Fuente: El Autor, 2013

3.2.7.11 TABLA TB_PRESTAMO

Describe los préstamos remitidos a los asociados.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_prestamo	character varying(10)	Si		Not Null	Identificador único de la tabla tb_prestamo
cod_cheque	character varying(10)		Si	Null	Identificador único de la tabla tb_cheque
id_usuario	character varying(10)		Si	Null	Identificador único de la tabla tb_usuario
cuotas	integer			Null	Número de cuotas
cuotas_restantes	integer			Null	Número de cuotas restantes
fecha_prestamo	date			Null	Número de cuotas restantes

Tabla 3-68: Descripción Tabla Tb_Prestamo

Fuente: El Autor, 2013

3.2.7.12 TABLA TB_RETENCION_PRESTAMO

Muestra el detalle de las retenciones de los préstamos efectuados.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_ret_pres	character varying(10)	Si		Not Null	Identificador único de la tabla tb_retención prestamos
id_usuario	character varying(10)		Si	Null	Identificador único de tabla usuario
cod_prestamo	character varying(10)		Si	Null	Identificador único de las tabla prestamo
monto_ret_pres	numeric(8,2)			Null	Monto de retención de préstamo
fecha_ret_pres	date			Null	Fecha de retención de préstamo

Tabla 3-69: Descripción Tabla Tb_Retencion_Prestamo

Fuente: El Autor, 2013

3.2.7.13 TABLA TB_RETENCION_APORTES

Muestra todas las retenciones de los aportes.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
id_usuario	character varying(10)			Null	Identificador único de la tabla usuario
cod_aporte	character varying(10)		Si	Null	Identificador único de la tabla aporte
monto_ret_apor	numeric(8,2)			Null	Monto de retención de aporte
fecha_ret_apor	date			Null	Fecha de retención de aporte
cod_ret_apor	integer	Si		Not Null	Identificador único de código de la retención de aporte

Tabla 3-70: Descripción Tabla Tb_Retencion_Aportes

Fuente: El Autor, 2013

3.2.7.14 TABLA TB_RETENCION_FAC

Muestra el detalle de las retenciones de los préstamos efectuados.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
cod_ret_fac	character varying(10)	Si		Not Null	Identificador único de código de la retención de la factura
id_usuario	character varying(10)		Si	Null	Identificador único de código de usuario
cod_fac_pag	character varying(10)		Si	Null	Identificador único de código de facturas por pagar
monto_ret_fac	numeric(8,2)			Null	Monto de retención de factura
fecha_ret_pres	date			Null	Fecha de retención de préstamo

Tabla 3-71: Descripción Tabla Tb_Retencion_Fac

Fuente: El Autor, 2013

3.2.7.15 TABLA TB_RETENCION_TOTAL

Muestra la retención total que se le descuenta al asociado,

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
id_usuario	character varying(10)		Si	Null	Identificador único de código de usuario
valor_total	numeric(8,2)			Null	Valor total de la retención
fecha_desde	date			Null	Fecha de inicio de la retención total
fecha_hasta	date			Null	Fecha de finalización de la retención total
fecha_retencion	date			Null	Fecha de retención
cod_retencion	integer	Si		Not Null	Identificador único de código de retención total

Tabla 3-72: Descripción Tabla Tb_Retencion_Total

Fuente: El Autor, 2013

3.2.7.16 TABLA TB_USUARIO

Almacena los datos informativos de los asociados de la presidencia.

Nombre	Tipo De Dato	Clave Primaria	Clave Foránea	Mandatorio	Descripción
id_usuario	character varying(10)	Si		Not Null	Identificador único de la tabla tb_usuario
cod_perfil	character varying(10)		Si	Null	Identificador único de la tb_perfil
apellidos	character varying(30)			Null	Descripción de los apellidos
nombres	character varying(30)			Null	Descripción de los nombres
password	character varying(25)			Null	Contraseña de usuario
dirección	character varying(80)			Null	Dirección
fono domicilio	character varying(15)			Null	Teléfono domiciliario
celular	character varying(12)			Null	Número de celular
servicio wap	character varying(2)			Null	Estado del servicio wap
nombramiento	character varying(2)			Null	Estado del nombramiento
sueldo	numeric(7,2)			Null	Valor del sueldo

Cargo	charactervar ying(80)			Null	Descripción del cargo
activo	charactervar ying(2)			Null	Estado del asociado

Tabla 3-73: Descripción Tabla Tb_Usuario

Fuente: El Autor, 2013

3.2.8 DISEÑO DE LA INTERFAZ Y NAVEGACIÓN

3.2.8.1 CONSTRUCCIÓN DEL SITIO WEB

En la construcción del sitio Web se detalla a breves rasgos todos los componentes de los que está compuesto el sistema de la Asociación de empleados de la Presidencia de la República del Ecuador (AsoPresi), tomando en cuenta que la estructura de la misma, varía en pequeñas características al momento de interactuar con el sistema.

Los elementos que posee la interfaz, permiten definir una estética de funcionalidad para que el usuario proceda de la mejor manera a con la interfaz y el proceso que deba realizar en si con características de monotonía y al mismo tiempo intuitivo.

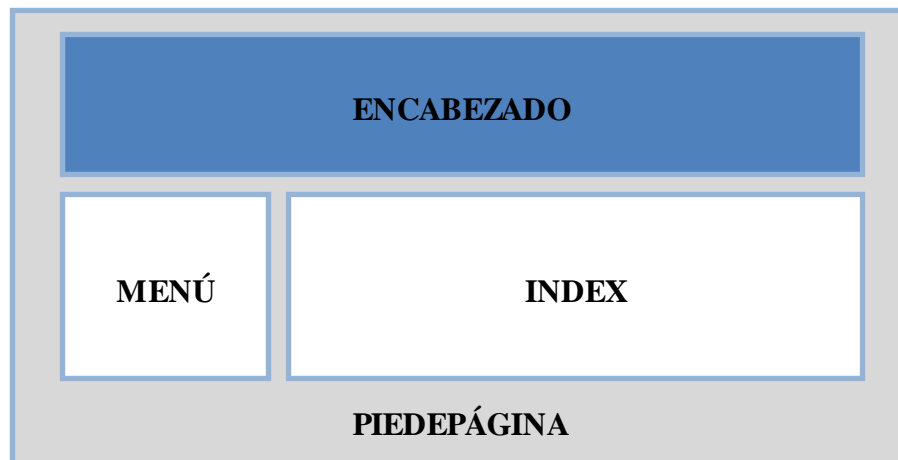


Figura 3-36 Diseño de Pantalla

Fuente: El Autor, 2013

3.2.8.2 ENCABEZADO



Figura 3-37 Sitio Web Encabezado
Fuente: El Autor, 2013

El encabezado o header, está ubicada en la parte superior de la página Web y contiene información propia del sistema a quien va dirigido el sistema, en la parte izquierda se ubica el logo y nombre del sitio, en este caso como logo se tiene al escudo del Ecuador junto con un nombre representativo, en la parte derecha del mismo se muestran imágenes de la presidencia de la república.

3.2.8.3 MENÚ



Figura 3-38 Sitio Web Menú
Fuente: El Autor, 2013

En el sistema AsoPresi como en la mayoría de aplicaciones, el menú permite la selección de una serie de opciones que el usuario puede elegir para realizar una determinada tarea.

Las opciones que presenta el menú, varía dependiendo de los diferentes perfiles de usuario que tiene la aplicación, como se muestra en la figura 3.38.

3.2.8.4 CONTENIDO



Figura 3-39 Sitio WebContenido
Fuente: El Autor, 2013

El contenido de la página es la parte del sistema que presenta a los usuarios información específica sobre una tarea específica, esta sección identifica e individualiza frente a las demás páginas del software.

Al ser la parte más importante de la interfaz, el espacio destinado a ella debe ser el mayor de todos, su ubicación siempre es central y en el sistema es la parte con la que el usuario tiene más interacción.

3.2.8.5 PIE DE PÁGINA



Figura 3-40 Sitio Web Pie de Página
Fuente: El Autor, 2013

El pie de página es un elemento de la interfaz de la aplicación, que contiene información muy concreta acerca del equipo de desarrollo del sistema.

Los contenidos del pie de página aparecerán centrados con respecto a los demás elementos de la página y solo son informativos por lo que su modificación no afecta al desempeño de la aplicación.

3.2.8.6 PLANTILLA WEB



Figura 3-41 Sitio Web Home
Fuente: El Autor, 2013

3.2.9 DISEÑO DE SOFTWARE

Las aplicaciones Web están organizadas siguiendo una arquitectura de 3 capas.

Capa presentación: implementada mediante páginas Web, tiene como objetivo interactuar, mostrar resultados y obtener datos del usuario para enviar la información necesaria a la siguiente capa.

Capa intermedia: constituye un verdadero núcleo de la aplicación Web, encargándose de procesar los datos del usuario, generar y enviar respuestas a la capa de presentación.

Capa de datos: almacena y recupera información del sistema.

Uno de los esquemas y, con toda seguridad, el más utilizado por los desarrolladores que utilizan J2EE, es la arquitectura Modelo Vista Controlador(MVC), la cual proporciona una clara separación entre las distintas responsabilidades de la aplicación.

En el siguiente gráfico se detalla como la aplicación AsoPresi se acoplará al modelo vista controlador, y de que componentes se combina con el fin de superar las limitaciones de otras arquitecturas, permitiendo incrementar el rendimiento, mantenimiento, reutilización y escalabilidad.

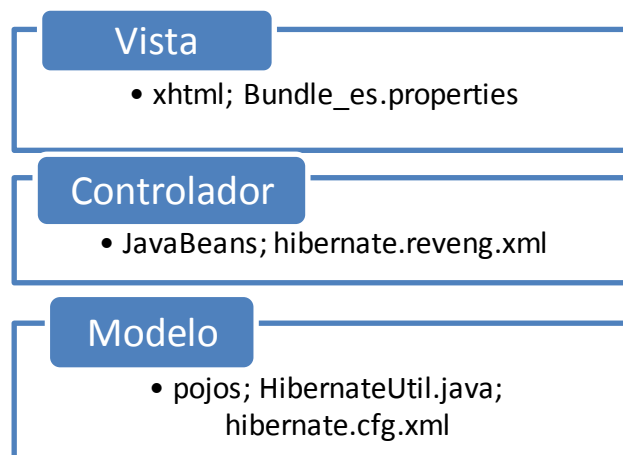


Figura 3-42 AsoPresi Modelo Vista Controlador
Fuente: El Autor, 2013

La programación por capas se define como un estilo de programación cuyo objetivo esencial es la separación del sistema en las capas que tenga definidas. De esta manera, si existe algún cambio, no es necesario interactuar con todas las capas y por tanto con todo el código.

CAPITULO IV

4 IMPLEMENTACIÓN Y PRUEBAS

Esta etapa consiste, en llevar a código fuente, todo lo diseñado en el capítulo anterior, siguiendo por completo los lineamientos impuestos en el diseño, además, se detallará las diferentes pruebas que se realizaron para la implementación

4.1 CÓDIGO FUENTE

El código fuente utilizado es como lo sugiere la metodología XP, reutilizado, es decir de una misma plantilla se harán los cambios necesarios para que la aplicación se adecue a los requerimientos.

4.1.1 PLANTILLA DE LA CAPA PRESENTACIÓN.

Para el siguiente caso se utilizarán etiquetas FACELETS, las mismas que vienen incluidas en JSF2.0

Facelets es un administrador de plantillas que permite diseñar y utilizar plantillas para JSF mediante el cual nos independizamos del estilo visual usado en las páginas, de forma que se permitirá en un futuro realizar cambios en el aspecto y los estilos de la aplicación de una forma fácil y elegante.

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:p="http://primefaces.org/ui"
      xmlns:f="http://java.sun.com/jsf/core">
<h:head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link href="/AsoPresi/faces/resources/css/cssLayout.css" rel="stylesheet" type="text/css" />
<title><ui:insert name="title">Home</ui:insert></title>
</h:head>
<h:body>
<div id="wrapper">
<div id="header">
```



```

<ui:insert name="top">
<style>
        .ui-panelgrid td {
border-width: 0px;
        }
        .ui-carousel .ui-carousel-header {
padding: 0px;
height: 0px;
margin: 0px;
        }
</style>

<p:carousel rows="1" autoPlayInterval="2000" pageLinks="1" circular="true" style="margin: 0px 0 0px 450px;">
<p:graphicImage value="../resources/css/images/palacio1.jpg" height="150"/>
<p:graphicImage value="../resources/css/images/palacio2.jpg" height="150" width="360"/>
<p:graphicImage value="../resources/css/images/palacio9.jpg" height="150" width="360"/>
</p:carousel>
</ui:insert>
</div>
<div id="page">
<div id="content">
<style>
        .ui-menu .ui-menuitem-link {
font-weight: 800;
        }
</style>
<h:form id="login-box">

<h:outputLabel value="#{blogin.lbUser}" style="font-weight: bold;color: #4f4da3 "/>
</h:form>
<p:separator/>
<br/>
<h:form>
<ui:insert name="left">
<p:growl id="mensajes" />
<p:tieredMenu model="#{blogin.model}" style="width: auto"/>
</ui:insert>
</h:form>
</div>
<div id="sidebar">
<div class="box">
<ui:insert name="content">Content</ui:insert>
</div>
</div>
</div>
<div id="footer">

```

```

<h:graphicImage value="../../resources/css/images/LOGO-UPS.png" width="100" style="margin: 50px 300px 0px;"/>
<br/>
        Copyright © 2012 Alejandro Hidalgo
<br/>
        Todos los derechos reservados
</div>
</div>
</h:body>
</html>

```

Figura 4-1 Código de Página Plantilla
Fuente: El Autor, 2013

4.1.2 BEANS.

El Bean asociado a la Figura 4-1, es Blogin.java que se muestra a continuación:

```

package controlador;

import java.util.ArrayList;
import java.util.List;
import javax.faces.bean.ManagedBean;
import javax.faces.bean.SessionScoped;
import javax.faces.context.FacesContext;
import java.util.ResourceBundle;
import javax.el.ExpressionFactory;
import javax.el.MethodExpression;
import javax.faces.event.ActionEvent;
import javax.faces.event.MethodExpressionActionListener;
import javax.faces.model.SelectItem;
import org.primefaces.component.menuitem.MenuItem;
import org.primefaces.component.submenu.Submenu;
import pojos.TbModulo;
import pojos.TbPerfil;
import pojos.TbUsuario;

/**
 *
 * @author Alejandro
 */
@ManagedBean
@SessionScoped
public class Blogin {

    private String cedula;
    private String clave;
    private boolean logeado;

```

```

private int contador;
private conexion con = new conexion();
private ArrayList<CargarMenu> menud;
private TbUsuario current;
    BUsuario usa;// = new BUsuario(sesion);
private MenuModel model;
private List<TbModulo> listperfil = new ArrayList<TbModulo>();
private String imgUser;
private String lblUser;
private String nomArchivo;

public Blogin() {
    }

public void cargarMenu(String perfil) {
    model = new DefaultMenuModel();

    //First submenu
    Submenu submenu = new Submenu();
    MenuItem item = new MenuItem();
    item.setValue("Usuario");
    if (perfil.equals("PER_03")) {
        item.setUrl("/faces/usuario/EditAso.xhtml");
        imgUser = "/AsoPresi/faces/resources/css/images/user1.png";
        lblUser = current.getApellidos() + " " + current.getNombres();
        nomArchivo = current.getIdUsuario() + current.getApellidos().replaceAll(" ", "");
    } else {
        if (perfil.equals("PER_02")) {
            item.setUrl("/faces/usuario/List.xhtml");
            imgUser = "/AsoPresi/faces/resources/css/images/secretaria.png";
            lblUser = "Secretaria";
        } else {
            if (perfil.equals("PER_01")) {
                item.setUrl("/faces/usuario/List.xhtml");
                imgUser = "/AsoPresi/faces/resources/css/images/admin1.png";
                lblUser = "Administrador";
            } else {
                item.setUrl("/faces/usuario/EditAso.xhtml");
                imgUser = "/AsoPresi/faces/resources/css/images/user1.png";
                lblUser = current.getApellidos() + " " + current.getNombres();
            }
        }
    }
    model.addMenuItem(item);

    listperfil = con.consultarHQL("from TbModulo where tbPerfil.codPerfil = '" + perfil + "' order by categoria");
    System.out.println("lista " + listperfil.size());
    submenu = new Submenu();
    submenu.setLabel(listperfil.get(0).getCategoria());

```

```

for (int i = 0; i < listperfil.size() - 1; i++) {
    item = new MenuItem();
    item.setValue(listperfil.get(i).getNombreModulo());
    item.setUrl(listperfil.get(i).getUrl());
    submenu.getChildren().add(item);
    if (!listperfil.get(i + 1).getCategoria().equals(listperfil.get(i).getCategoria())) {
        model.addSubmenu(submenu);
        submenu = new Submenu();
        submenu.setLabel(listperfil.get(i + 1).getCategoria());
    }
}

}

}

public void validar() {

    List list_acceso = con.consultarHQL("from TbUsuario as us where us.idUsuario = '" + cedula
+ "'" and us.password = '" + clave + "'");

    if (list_acceso != null && list_acceso.size() > 0) {
        try {
            current = (TbUsuario) list_acceso.get(0);
            System.out.println("tb." + current.getTbPerfil().getCodPerfil());

            usa = new BUsuario(current);

            FacesContext.getCurrentInstance().getExternalContext().redirect("../index.xhtml");

            FacesContext.getCurrentInstance().getExternalContext().getSessionMap().put("nombre_usua", cedula);
            JsUtil.addSuccessMessage(current.getNombres() + " " + current.getApellidos());
            FacesMessage facesMsg = new FacesMessage(FacesMessage.SEVERITY_INFO,
            ResourceBundle.getBundle("/resources/Bundle_es").getString("LoginCorrecto"),
            ResourceBundle.getBundle("/resources/Bundle_es").getString("LoginCorrecto"));

            FacesContext.getCurrentInstance().addMessage("successInfo", facesMsg);
            cargarMenu(current.getTbPerfil().getCodPerfil().toString());
            logeado = true;
        } catch (Exception e) {
            System.out.println("Vino al catch");
        }
        } else {
            JsUtil.addErrorMessage(ResourceBundle.getBundle("/resources/Bundle_es").getString("LoginIncorrec
cto"));
        }
    }

    //Metodo para cerrar sesion
    public String sesionIndex() {
        FacesContext.getCurrentInstance().getExternalContext().invalidateSession();
        return "/login/Login.xhtml";
    }
}

```

```
--getters y setters correspondientes.
}
```

Figura 4-2 Código Blogin.Java
Fuente: El Autor, 2013

4.1.3 MANEJO DE WEB SERVICE

Para la obtención y exportación de datos se utiliza Web services, las clases que permiten el manejo de estos son similares para cada una de las fases del proceso y se describe de la siguiente manera:

Se especifica la clase ServiceMensual del Web services en la cual se crea los objetos de la tabla tb_retencion_total, para ser exportados con la información de los asociados.

```
package modelo;

import conexion.conectar;
import pojos.RetencionesTotales;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.List;
import javax.annotation.Resource;
import javax.ejb.LocalBean;
import javax.ejb.Stateless;
import javax.jws.WebMethod;
import javax.jws.WebParam;
import javax.jws.WebService;
import javax.servlet.jsp.tagext.TryCatchFinally;
import javax.sql.DataSource;

/**
 *
 * @author Alejandro
 */
```

```

@WebService
@Stateless
@LocalBean
public class ServiceMensual {

    private Connection ObtenerConnection2() {
        Connection conn = null;
        try{
            conectar conn1 = new conectar();
            conn = conn1.Conectar();
        } catch (Exception e) {
        }
        return conn;
    }

    /**
     * Web service operation
     */

    @WebMethod(operationName = "buscarRetenciones")
    public java.util.List<pojos.RetencionesTotales> buscarRetenciones(@ WebParam(name =
"fechaBusqueda") String fechaBusqueda) {

        List<RetencionesTotales> result = new ArrayList<RetencionesTotales>(0);
        Connection conn = null;
        try{
            conn = ObtenerConnection2();
            Statement sta = conn.createStatement();
            PreparedStatement preSta = conn.prepareStatement(RetencionesTotales.SELECT_FECHA2);
            SimpleDateFormat dateParser = new SimpleDateFormat("dd/MM/yyyy");
            java.util.Date d = dateParser.parse(fechaBusqueda);
            java.sql.Date sqlDate = new java.sql.Date(d.getTime());
            preSta.setDate(1, sqlDate);
            ResultSet rset = preSta.executeQuery();
            //ResultSet rset = sta.executeQuery(RetencionesTotales.SELECT_FECHA);
            while(rset.next()) {
                RetencionesTotales retencion = new RetencionesTotales();

                retencion.setCod_retencion(rset.getInt("cod_retencion"));
                retencion.setId_usuario(rset.getString("id_usuario"));
                retencion.setValor_total(rset.getBigDecimal("valor_total"));
                retencion.setFecha_desde(rset.getDate("fecha_desde"));
                retencion.setFecha_hasta(rset.getDate("fecha_hasta"));
                retencion.setFecha_retencion(rset.getDate("fecha_retencion"));

                result.add(retencion);
            }
            rset.close();
            sta.close();
        } catch (Exception e) {

```

```

e.printStackTrace();
    } finally {
try{
if(conn != null) {
conn.close();
    }
    } catch (Exception e) {
e.printStackTrace();
    }
    }
return result;
    }
}

```

Figura 4-3 Código Web Service: Servicemensual.Java
Fuente: El Autor, 2013

4.1.4 CAPA ANDROID

El código que se usa para que se pueda mostrar en un dispositivo móvil, es xml para que pueda presentarse como formulario.

Se presenta el archivo: préstamo_activity.xml y forma de cómo se va construyendo la interfaz de usuario final.

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
        <TableRow
            android:id="@+id/tableRow1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >
            <TextView
                android:id="@+id/tvUsuario"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="TextView" />
            </TableRow>
        <TableRow

```

```

android:id="@+id/tableRow2"
android:layout_width="wrap_content"
android:layout_height="wrap_content" >
</TableRow>
<TableRow
android:id="@+id/tableRow3"
android:layout_width="wrap_content"
android:layout_height="wrap_content" >
</TableRow>
<TableRow
    android:id="@+id/tableRow4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >
</TableRow>
</TableLayout>
</LinearLayout>

```

Figura 4-4 Código Android Prestamo_Activity.Xml
Fuente: El Autor, 2013

4.2 MANUALES

Los Manuales se ha decidido ubicarlos en una sección separada para que puedan ser consultados por parte de las personas interesadas en el uso del software, en el ANEXO B, se puede apreciar a detalle los manuales con los diferentes requerimientos y perfiles.

4.3 PRUEBAS

En la cadena de valor del desarrollo de un software específico, el proceso de pruebas es fundamental antes de que sea implementado con totalidad para de esta manera detectar errores o fallas, conceptos como estabilidad, escalabilidad, eficiencia y seguridad a la calidad de un producto bien desarrollado.

Las pruebas realizadas para cumplir el objetivo son las siguientes:

- Pruebas de Caja Negra
- Pruebas de Caja Blanca

- Pruebas del Sistema

4.3.1 PRUEBAS DE CAJA NEGRA

Las pruebas de caja negra también llamadas:

- pruebas de caja opaca
- pruebas funcionales
- pruebas de entrada/salida
- pruebas inducidas por los datos

La característica principal de las pruebas de caja negra es cuando prescinde del código y se enfoca a lo que se muestra en la capa presentación, lo que trata de examinar es casos y circunstancias en los que el módulo no hace lo que se espera de él.- Están especialmente demostradas en aquellos módulos que van a ser interfaz con el usuario (en sentido general: teclado, pantalla, ficheros, canales de comunicaciones, etc.)

Las pruebas de caja negra intentan encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Antecedentes.

Se tiene como premisa que los mensajes mostrados en pantalla (pop-up), tendrán el siguiente formato:

- Mensajes de error de color rojo.
- Mensajes de información de color azul.

- Cualquier mensaje es mostrado como pop-up en la parte superior derecha, el cual se disolverá en 3 segundos.
- Cualquier mensaje se mostrará en una etiqueta y se ubicará en la parte superior de la pantalla.

Interfaz: Iniciar Sesión.

Número:	1			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Al presionar el botón “LOGIN”, sin datos ingresados	Presenta un mensaje de error “ <i>El campo usuario es requerido, El campo password es requerido</i> ”	1	SI
2	El campo usuario tiene como tamaño máximo 10 caracteres.	No permite ingresar más caracteres que el número establecido (10 caracteres)	1	SI
3	El campo password se muestra con *** (asteriscos).	El campo se muestra protegido para que no se pueda ver la contraseña.	1	SI
4	Ingresar usuario y/o password incorrecta Presionando el botón “LOGIN ”	Presenta un mensaje de error “ <i>El usuario o password son incorrectos</i> ”	1	SI
Conclusión:	La interfaz no permite el acceso al sistema, cuando los datos son incorrectos o no existen en la base de datos.			
Observación:	En caso de ocurrir un error, la pantalla muestra un mensaje de error, eliminando la contraseña para ingresar los datos nuevamente.			

Tabla 4-1 Prueba de Caja Negra: Iniciar Sesión, Datos Incorrectos
Fuente: El Autor, 2013

Número:	2			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Ingresar usuario y password correcto de un administrador Presionando el botón “LOGIN ”	Acceso a la interfaz de Administrador, en la parte izquierda superior al menú un ícono azul y una etiqueta: “ <i>Administrador</i> ”, lo cual	1	SI

		nos muestra con que usuario se ingresó al sistema		
2	Ingresar usuario y password correcto de una secretaria Presionando el botón “ <i>LOGIN</i> ”	Acceso a la interfaz de Secretaria, en la parte izquierda superior al menú un ícono azul y una etiqueta: “ <i>Secretaria</i> ”, con su respectivo menú.	1	SI
3	Ingresar usuario y password correcto de un asociado Presionando el botón “ <i>LOGIN</i> ”	Acceso a la interfaz de Asociado, en la parte izquierda superior al menú un ícono azul y una etiqueta: “ <i>Apellidos y Nombres</i> ”, lo cual nos muestra con que usuario se ingresó al sistema y su respectivo menú.	1	SI
Conclusión:	La interfaz permite el acceso al sistema, cuando los datos son correctos.			
Observación:	Solo se puede ingresar al sistema con usuarios y password validos en la base de datos.			

Tabla 4-2 Prueba de Caja Negra: Iniciar Sesión, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Ingreso de Usuarios.

Número:	3			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar usuarios en la lista de usuarios, ingresando datos que no existen en el campo: cédula y/o Apellidos y/o Nombres.	Presenta un mensaje de error “ <i>No hay registros encontrados</i> ”	1	SI
2	Dar clic en el ícono Nuevo Usuario.	Re direcciona a la página Nuevo Usuario.	1	SI
3	Si los datos no son llenados o se los deja en blanco para después dar clic en “guardar”	Muestra un mensaje de error: “Cédula obligatoria; Apellidos son requeridos; Nombres requeridos.”	1	SI
4	Al dar clic en el campo Cédula	Presenta un mensaje de error “Cédula Obligatoria “	1	SI

5	Los campos por defecto son: Nombramiento, Activo y Perfil	Nombramiento valor: NO, Activo valor: SI, Perfil valor: depende el perfil.	1	SI
6	El campo perfil depende si es administrador o secretaria.	Perfil mostrado para administrador: Administrador, Secretaria y Asociado. Perfil mostrado para secretaria: Asociado	1	SI
7	El campo password no es llenado al momento de crear un nuevo usuario.	El campo password está deshabilitado.	1	SI
8	Si el campo perfil esta seleccionado como "ASOCIADO", ingresamos una cédula errónea.	Muestra un mensaje: "La cédula es incorrecta"	1	SI
9	Al llenar todos los campos pero con un usuario que ya existe.	Muestra un mensaje: "Error, La información no fue guardada"	1	SI
Conclusión:	La interfaz no nos permite guardar un usuario mientras no cumpla con todos los parámetros establecidos.			
Observación:	En caso de ocurrir un error de validación, existe un ícono basurero que permitirá limpiar el contenido de la pantalla.			

Tabla 4-3 Prueba de Caja Negra: Ingreso de Usuarios, Datos Incorrectos
Fuente: El Autor, 2013

Número:	4			
Caso:	Datos correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar usuarios en la lista de usuarios, ingresando datos que existen en el campo: cédula y/o Apellidos y/o Nombres.	Presenta el registro obtenido de la búsqueda.	1	SI
2	Clic en el registro buscado y posteriormente clic en "Editar"	Muestra el registro seleccionado en una nueva pantalla llamada Editar Usuario	1	SI
3	Clic en el ícono: "Restablecer Contraseña"	Muestra un mensaje de error: "La contraseña fue reiniciada correctamente."	1	SI
4	Al modificar un campo del usuario y luego	Muestra un mensaje de error: "El usuario fue	1	SI

	presionar el ícono “Actualizar”	actualizado correctamente.”		
5	Si es necesario eliminar un usuario clic en “Eliminar Usuario”	Muestra un mensaje de error: “El usuario fue eliminado correctamente.” y la página se muestra en blanco	1	SI
6	Llenar todos los campos para un Nuevo Usuario	Muestra un mensaje de error: “El usuario fue creado exitosamente.”	1	SI
Conclusión:	La interfaz nos permite guardar un usuario mientras cumpla con los parámetros establecidos.			
Observación:	Si se ingresa un usuario con perfil asociado valida que la cédula sea correcta.			

Tabla 4-4 Prueba de Caja Negra: Ingreso de Usuarios, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Ingreso de Cheques.

Número:	5			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar cheques en la lista, ingresando datos que no existen en el campo: NumCheque y/o FechaEmisión y/o ValorCheque.	Presenta un mensaje de error “ <i>No hay registros encontrados</i> ”	1	SI
2	Dar clic en el ícono Nuevo Cheque.	Re direcciona a la página Nuevo Cheque.	1	SI
3	Si los datos no son llenados o se los deja en blanco para después dar clic en “guardar”	Muestra un mensaje de error: “El nombre del banco es obligatorio; numCheque se necesita un valor; ValorCheque se necesita un valor.”	1	SI
4	Al llenar el campo ValorCheque con caracteres que no sean numéricos	Presenta un mensaje de error “msj debe ser un número decimal”	1	SI
Conclusión:	La interfaz no nos permite guardar un cheque mientras los campos a validar no sean correctos.			

Tabla 4-5 Prueba de Caja Negra: Ingreso de Cheques, Datos Incorrectos
Fuente: El Autor, 2013

Número:	6			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar cheques en la lista, ingresando datos que existen en el campo: NumCheque y/o FechaEmisión y/o ValorCheque.	Presenta el registro obtenido de la búsqueda	1	SI
2	Al llenar los campos necesarios para un nuevo cheque	Presenta un mensaje “El cheque fue agregado correctamente”	1	SI
3	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar Cheque	1	SI
4	Al modificar un campo del cheque y luego presionar el ícono “Actualizar”	Muestra el mensaje: “El cheque fue actualizado correctamente.”	1	SI
5	Si es necesario eliminar un cheque clic en “Eliminar Cheque”	Muestra el mensaje: “El cheque fue eliminado correctamente.” y la página se muestra en blanco	1	SI
Conclusión:	La interfaz nos permite guardar un cheque mientras los campos a validar sean correctos.			
Observación:	Un cheque es necesario para que se puedan realizar los pagos de facturas y préstamos, además los cheques utilizados no se muestran en los menús desplegables de páginas que necesiten de cheques.			

Tabla 4-6 Prueba de Caja Negra: Ingreso de Cheques, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Ingreso de Casa Comercial.

Número:	7			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar casas comerciales en la lista, ingresando datos que no existen en el campo: NombreCC	Presenta un mensaje de error “No hay registros encontrados”	1	SI
2	Dar clic en el ícono Nueva Casa Comercial.	Re direcciona a la página Nuevo Casa Comercial.	1	SI
3	Si los datos no son	Muestra un mensaje de	1	SI

	llenados o se los deja en blanco para después dar clic en “guardar”	error: “El nombre casa comercial es un valor necesario”		
Conclusión:	La interfaz no nos permite guardar una casa comercial mientras los campos a validar no sean correctos.			
Observación:	En caso de ocurrir un error de validación, existe un ícono basurero que permitirá limpiar el contenido de la pantalla.			

Tabla 4-7 Prueba de Caja Negra: Ingreso de Casa Comercial, Datos Incorrectos
Fuente: El Autor, 2013

Número:	8			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar casas comerciales en la lista, ingresando datos que existen en el campo: NombreCc.	Presenta el registro obtenido de la búsqueda	1	SI
2	Al llenar los campos necesarios para una nueva casa comercial	Presenta un mensaje “La casa comercial fue agregada correctamente”	1	SI
3	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar Casa Comercial	1	SI
4	Al modificar un campo de la casa comercial y luego presionar el ícono “Actualizar”	Muestra el mensaje: “La casa comercial fue actualizada correctamente.”	1	SI
5	Si es necesario eliminar una casa comercial clic en “Eliminar Casa Comercial”	Muestra el mensaje: “La casa comercial fue eliminada correctamente.”	1	SI
Conclusión:	La interfaz nos permite guardar una casa comercial mientras los campos a validar sean correctos.			
Observación:	Al guardar una casa comercial la pantalla se limpia y muestra un mensaje de éxito en la operación			

Tabla 4-8 Prueba de Caja Negra: Ingreso de Casa Comercial, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Ingreso de Facturas

Número:	9			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito

1	Filtrar facturas en la lista, ingresando datos que no existen en los campos	Presenta un mensaje de error “No hay registros encontrados”	1	SI
2	Dar clic en el ícono Nueva Factura.	Re direcciona a la página Nuevo Factura.	1	SI
3	Si los datos no son llenados o se los deja en blanco para después dar clic en “guardar”	Muestra un mensaje de error: “La casa comercial es un valor necesario; El NumCheque es un valor requerido; Total a pagar es un valor requerido”	1	SI
4	Si los datos en el campo TotalPagar, sub, iva, montoFactura no son números	Muestra un mensaje de error: “debe ser un número decimal”	1	SI
Conclusión:	La interfaz no nos permite guardar valores que no sean numéricos en campos que así lo requieren.			
Observación:	Al guardar una factura esta se guarda con una clave primaria que no permite tener valores duplicados.			

Tabla 4-9 Prueba de Caja Negra: Ingreso de Factura, Datos Incorrectos
Fuente: El Autor, 2013

Número:	10			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar facturas en la lista, ingresando datos que existen en los campos.	Presenta el registro obtenido de la búsqueda y muestra un panel detallando la factura seleccionada.	1	SI
2	Al llenar los campos necesarios para una nueva factura	Presenta un mensaje “La factura fue agregada correctamente”	1	SI
3	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar Factura	1	SI
4	Al modificar un campo de la factura y luego presionar el ícono “Actualizar”	Muestra el mensaje: “La factura fue actualizada correctamente.”	1	SI
5	Si es necesario eliminar una factura clic en “Eliminar Factura”	Muestra el mensaje: “La factura fue eliminada correctamente.”	1	SI
Conclusión:	La interfaz nos permite guardar una factura mientras los campos a validar sean correctos.			

Observación:	Si es necesario eliminar una factura esta no debe tener relación con otras pantallas por ejemplo retenciones de facturas.
--------------	---

Tabla 4-10 Prueba de Caja Negra: Ingreso de Factura, Datos Correctos

Fuente: El Autor, 2013

Interfaz: Ingreso de Préstamos

Número:	11			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar los préstamos en la lista, ingresando datos que no existen en los campos: Cédula, ValorCheque, FechaPrestamo	Presenta un mensaje de error <i>"No hay registros encontrados"</i>	1	SI
2	Dar clic en el ícono Nuevo Préstamo.	Re direcciona a la página Nuevo Préstamo.	1	SI
3	Si los datos no son llenados o se los deja en blanco para después dar clic en "guardar"	Muestra un mensaje de error: "El asociado es requerido; Número de Cheque requerido; Número de Cuotas requerido; fecha de préstamo necesita un valor"	1	SI
4	Si los datos en el campo Cuotas no son números	Muestra un mensaje de error: "debe ser un número formado por uno o varios dígitos"	1	SI
Conclusión:	La interfaz no nos permite guardar valores que no sean numéricos en campos que así lo requieren.			
Observación:	Al guardar un préstamo esta se guarda con una clave primaria que no permite tener valores duplicados.			

Tabla 4-11 Prueba de Caja Negra: Ingreso de Préstamo, Datos Incorrectos

Fuente: El Autor, 2013

Número:	12			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar préstamo en la lista, ingresando datos que existen en los campos: Cédula, ValorCheque, FechaPrestamo.	Presenta el registro obtenido de la búsqueda.	1	SI

2	Al llenar los campos necesarios para un nuevo préstamo	Presenta un mensaje “El préstamo fue agregado correctamente”	1	SI
3	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar Préstamo	1	SI
4	Al modificar un campo de la préstamo y luego presionar el ícono “Actualizar”	Muestra el mensaje: “El préstamo fue actualizado correctamente.”	1	SI
5	Si es necesario eliminar una préstamo clic en “Eliminar Préstamo”	Muestra el mensaje: “El préstamo fue eliminado correctamente.”	1	SI
Conclusión:	La interfaz permite guardar un préstamo siempre que se haya creado primero un cheque y este no esté relacionado con otras interfaces.			
Observación:	Si es necesario eliminar un préstamo esta no debe tener relación con otras pantallas por ejemplo retenciones de facturas.			

Tabla 4-12 Prueba de Caja Negra: Ingreso de Préstamo, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Ingreso de Aportes

Número:	13			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Exito
1	Filtrar los aportes en la lista, ingresando datos que no existen en los campos: Nombre Aporte, Monto	Presenta un mensaje de error “No hay registros encontrados”	1	SI
2	Dar clic en el ícono Nuevo Aporte.	Re direcciona a la página Nuevo Aporte.	1	SI
3	Si los datos no son llenados o se los deja en blanco para después dar clic en “guardar”	Muestra un mensaje de error: “Nombre de Aporte es requerido, Monto necesita un valor”	1	SI
4	Si los datos en el campo Monto no son números	Muestra un mensaje de error: “debe ser un número decimal”	1	SI
Conclusión:	La interfaz no nos permite guardar valores que no sean numéricos en campos que así lo requieren.			
Observación:	Los aportes permiten escoger de un menú los meses que será vigente, o si está activo o no y para quienes se aplican los aportes.			

Tabla 4-13 Prueba de Caja Negra: Ingreso de Aportes, Datos Incorrectos
Fuente: El Autor, 201

Número:	14			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Filtrar aportes en la lista, ingresando datos que existen en los campos: Nombre Aporte, Monto.	Presenta el registro obtenido de la búsqueda.	1	SI
2	Al llenar los campos necesarios para un nuevo aporte	Presenta un mensaje “El aportes fue agregado correctamente”	1	SI
3	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar Aporte	1	SI
4	Al modificar el campo Activo y luego presionar el ícono “Actualizar”	El aporte deja de estar activo o vuelve a estar activo.	1	SI
5	Al modificar el campo Meses de Aporte y luego presionar el ícono “Actualizar”	Nos permite controlar el número de meses que deseamos que se descuenta el aporte	1	SI
6	Al modificar el campo Meses Restantes y luego presionar el ícono “Actualizar”	Nos permite controlar el número de meses que deseamos que se descuenta el aporte	1	SI
7	Al modificar un campo del aporte y luego presionar el ícono “Actualizar”	Muestra el mensaje: “El aporte fue actualizado correctamente.”	1	SI
8	Si es necesario eliminar un aporte clic en “Eliminar Aporte”	Muestra el mensaje: “El aporte fue eliminado correctamente.”	1	SI
Conclusión:	Los aportes son controlados al editar un aporte.			
Observación:	La interfaz nos permite guardar un aporte y controlar cuantas veces se descontarán, inhabilitarle o aumentar o disminuir los meses de descuento.			

Tabla 4-14 Prueba de Caja Negra: Ingreso de Aportes, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Retenciones de Aportes

Número:	15			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito

1	Perfil Secretaria: Clic en el ícono buscar sin haber seleccionado una fecha de retenciones de aportes	Presenta un mensaje de error " <i>Seleccione la fecha para generar la búsqueda</i> " y la menú de las fechas se muestra de color rojo	1	SI
2	Al buscar en la lista si no se ingresan los datos correctos para la búsqueda.	Muestra un mensaje de error: "No hay registros encontrados"	1	SI
3	Si las retenciones ya han sido generadas para ese mes y se presiona el ícono de generar retenciones de aportes	Presenta un mensaje de error " <i>Las retenciones de aportes ya han sido generadas para el mes actual: fecha_actual</i> "	1	SI
4	Clic en el ícono Eliminar sin haber seleccionado una fecha de retenciones de aportes	Muestra un mensaje de error: " <i>Seleccione la fecha de las retenciones de aportes a eliminar</i> "	1	SI
5	Filtrar los aportes en la lista, ingresando datos que no existen en los campos:	Presenta un mensaje de error " <i>No hay registros encontrados</i> "	1	SI
6	Dar clic en el ícono Nueva Retención de Aporte.	Re direcciona a la página Nueva Retención de Aporte.	1	SI
7	Si los datos no son llenados o se los deja en blanco para después dar clic en "guardar"	Muestra un mensaje de error: " <i>Usuario es requerido, Aporte es requerido, Monto es requerido</i> "	1	SI
8	Si los datos en el campo Monto no son números	Muestra un mensaje de error: "debe ser un número decimal"	1	SI
Conclusión:	La interfaz no nos permite guardar valores que no sean numéricos en campos que así lo requieren.			
Observación:	La interfaz nos permite generar retenciones de aportes y controlando la información necesaria para validar su ingreso.			

Tabla 4-15 Prueba de Caja Negra: Retenciones de Aportes, Datos Incorrectos
Fuente: El Autor, 2013

Número:	16			
Caso:	Datos Correctos			
Nº	Acción	Resultado Esperado	Intentos	Éxito

1	Clic en el ícono buscar después de haber seleccionado una fecha de retenciones de aportes	Muestra los registros que corresponden a esa fecha	1	SI
2	Filtrar retenciones de aportes en la lista, ingresando datos que existen en los campos.	Presenta el registro obtenido de la búsqueda.	1	SI
3	Al llenar los campos necesarios para un nueva retención aporte	Presenta un mensaje “El aportes fue agregado correctamente”	1	SI
4	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar Retención Aporte	1	SI
5	Al modificar un campo de la retención aporte y luego presionar el ícono “Actualizar”	Muestra el mensaje: “El aporte fue actualizado correctamente.”	1	SI
6	Seleccionar una fecha del menú y presionar el botón eliminar	Muestra un mensaje “Las retenciones han sido eliminadas correctamente”	1	SI
7	Las fechas que se muestran en el menú desplegable	Solo son las fechas del mes actual	1	SI
8	Clic en generar retenciones de aportes	Se generan las retenciones y con la fecha actual y muestra un mensaje “Las retenciones han sido creadas exitosamente”	1	SI
Conclusión:	Para editar una retención la fecha no debe ser mayor a la fecha actual por 15 días, porque es el tiempo que ya fueron enviados a la presidencia y no se pueden editar ni eliminar.			
Observación:	Las retenciones solo se pueden crear una vez al mes. Las retenciones de aportes solo están habilitadas para el perfil administrador, el cual puede crear y editar.			

Tabla 4-16 Prueba de Caja Negra: Retenciones de Aportes, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Retenciones de Préstamos

Número:	17			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito

1	Perfil Secretaria: Clic en el ícono buscar sin haber seleccionado una fecha de retenciones de aportes	Presenta un mensaje de error " <i>Seleccione la fecha para generar la búsqueda</i> " y la menú de las fechas se muestra de color rojo	1	SI
2	Al buscar en la lista si no se ingresan los datos correctos para la búsqueda.	Muestra un mensaje de error: "No hay registros encontrados"	1	SI
3	Si las retenciones ya han sido generadas para ese mes y se presiona el ícono de generar retenciones de préstamos	Presenta un mensaje de error " <i>Las retenciones de facturas ya han sido generadas para el mes actual: fecha_actual</i> "	1	SI
4	Clic en el ícono Eliminar sin haber seleccionado una fecha de retenciones de préstamos	Muestra un mensaje de error: " <i>Seleccione la fecha de las retenciones de préstamos a eliminar</i> "	1	SI
5	Filtrar los aportes en la lista, ingresando datos que no existen en los campos:	Presenta un mensaje de error " <i>No hay registros encontrados</i> "	1	SI
6	Dar clic en el ícono Nueva Retención de Préstamos.	Re direcciona a la página Nueva Retención de Préstamo.	1	SI
7	Si los datos no son llenados o se los deja en blanco para después dar clic en "guardar"	Muestra un mensaje de error: " <i>Usuario es requerido, Préstamo es requerido, Monto es requerido</i> "	1	SI
8	Si los datos en el campo Monto no son números	Muestra un mensaje de error: "debe ser un número decimal"	1	SI
Conclusión:	La interfaz no nos permite guardar valores que no sean numéricos en campos que así lo requieren.			
Observación:	La interfaz nos permite generar retenciones de préstamos y controlando la información necesaria para validar su ingreso.			

Tabla 4-17 Prueba de Caja Negra: Retenciones de Préstamos, Datos Incorrectos
Fuente: El Autor, 2013

Número:	18			
Caso:	Datos Correctos			
Nº	Acción	Resultado Esperado	Intentos	Éxito

1	Clic en el ícono buscar después de haber seleccionado una fecha de retenciones de préstamos	Muestra los registros que corresponden a esa fecha	1	SI
2	Filtrar retenciones de aportes en la lista, ingresando datos que existen en los campos.	Presenta el registro obtenido de la búsqueda.	1	SI
3	Al llenar los campos necesarios para un nueva retención aporte	Presenta un mensaje “El aportes fue agregado correctamente”	1	SI
4	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar RetenciónPréstamo	1	SI
5	Al modificar un campo de la retenciónpréstamo y luego presionar el ícono “Actualizar”	Muestra el mensaje: “El préstamo fue actualizado correctamente.”	1	SI
6	Seleccionar una fecha del menú y presionar el botón eliminar	Muestra un mensaje “Las retenciones han sido eliminadas correctamente”	1	SI
7	Las fechas que se muestran en el menú desplegable	Solo son las fechas del mes actual	1	SI
8	Clic en generar retenciones de préstamos	Se generan las retenciones y con la fecha actual y muestra un mensaje “Las retenciones han sido creadas exitosamente”	1	SI
Conclusión:	Para editar una retención la fecha no debe ser mayor a la fecha actual por 15 días, porque es el tiempo que ya fueron enviados a la presidencia y no se pueden editar ni eliminar.			
Observación:	Las retenciones solo se pueden crear una vez al mes. Las retenciones de préstamos solo están habilitadas para el perfil administrador, el cual puede crear y editar.			

Tabla 4-18 Prueba de Caja Negra: Retenciones de Préstamos, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Retenciones de Facturas

Número:	19			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito

1	Perfil Secretaria: Clic en el ícono buscar sin haber seleccionado una fecha de retenciones de aportes	Presenta un mensaje de error “Ingrese los rango de fecha desde hasta” y los campos de las fechas se muestra de color rojo	1	SI
2	Al buscar en la lista si no coinciden las fechas de búsqueda.	Muestra un mensaje de error: “No hay registros encontrados”	1	SI
3	Si los datos no son llenados o se los deja en blanco para después dar clic en “guardar”	Muestra un mensaje de error: “ <i>Usuario es requerido, Factura es requerido, Monto es requerido</i> ”	1	SI
4	Filtrar los aportes en la lista, ingresando datos que no existen en los campos:	Presenta un mensaje de error “ <i>No hay registros encontrados</i> ”	1	SI
5	Dar clic en el ícono Nueva Retención de Factura.	Re direcciona a la página Nueva Retención de Préstamo.	1	SI
6	Si los datos en el campo Monto no son números	Muestra un mensaje de error: “debe ser un número decimal”	1	SI
Conclusión:	La interfaz no nos permite guardar valores que no sean numéricos en campos que así lo requieren.			
Observación:	La interfaz nos permite generar retenciones de facturas y controlando la información necesaria para validar su ingreso.			

Tabla 4-19 Prueba de Caja Negra: Retenciones de Facturas, Datos Incorrectos

Fuente: El Autor, 2013

Número:	20			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Clic en el ícono buscar después de haber seleccionado una fecha desde hasta de retenciones de facturas	Muestra los registros que corresponden a esa fecha	1	SI
2	Filtrar retenciones de facturas en la lista, ingresando datos que existen en los campos.	Presenta el registro obtenido de la búsqueda.	1	SI
3	Al llenar los campos necesarios para una nueva retenciónfactura	Presenta un mensaje “La retención fue agregada correctamente” y muestra en una lista de las retenciones que se han almacenado para ese	1	SI

		usuario y esa factura		
4	Clic en el registro buscado y posteriormente clic en "Editar"	Muestra el registro seleccionado en una nueva pantalla llamada Editar RetenciónFactura	1	SI
5	Al modificar un campo de la retenciónfactura y luego presionar el ícono "Actualizar"	Muestra el mensaje: "La retención fue actualizada correctamente."	1	SI
6	Seleccionar retención y presionar el botón eliminar	Muestra un mensaje "La retención ha sido eliminada correctamente"	1	SI
Conclusión:	Las retenciones de facturas corresponden a los gastos que se tiene con las diferentes casas comerciales.			
Observación:	Las retenciones de facturas muestran en un panel las retenciones para una misma casa comercial por usuario, ayuda a saber cómo se va almacenado una factura a un determinado usuario.			

Tabla 4-20 Prueba de Caja Negra: Retenciones de Facturas, Datos Correctos
Fuente: El Autor, 2013

Interfaz: Retenciones Totales

Número:	21			
Caso:	Datos Incorrectos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Clic en el ícono "generar retenciones" sin haber ingresado las fechas de corte	Presenta un mensaje de error " <i>Ingrese los rango de fecha desde hasta</i> " y los campos de las fechas se muestra de color rojo	1	SI
2	Clic en el ícono buscar sin haber seleccionado una fecha del menú	Muestra un mensaje de error: " <i>Seleccione la fecha para generar la búsqueda</i> "	1	SI
3	Filtrar las retenciones en la lista, ingresando datos que no existen en los campos:	Presenta un mensaje de error " <i>No hay registros encontrados</i> "	1	SI
4	Clic en el "ícono generar retenciones", si las retenciones ya fueron generadas para esas fechas	Muestra un mensaje de error: " <i>Las retenciones ya han sido generadas para las siguientes fechas</i> "	1	SI
Conclusión:	La interfaz no nos permite generar retenciones que hayan sido creadas con fechas similares.			
Observación:	Las retenciones totales solo se generan una vez al mes.			

Tabla 4-21 Prueba de Caja Negra: Retenciones Totales, Datos Incorrectos
Fuente: El Autor, 2013

Número:	22			
Caso:	Datos Correctos			
N°	Acción	Resultado Esperado	Intentos	Éxito
1	Clic en el ícono buscar después de haber seleccionado una fecha del menú desplegable	Muestra los registros que corresponden a esa fecha	1	SI
2	Filtrar retenciones de totales en la lista, ingresando datos que existen en los campos.	Presenta el registro obtenido de la búsqueda.	1	SI
3	Al llenar los campos necesarios para un nueva retención total	Presenta un mensaje “ <i>La retención fue agregada correctamente</i> ”	1	SI
4	Clic en el registro buscado y posteriormente clic en “Editar”	Muestra el registro seleccionado en una nueva pantalla llamada Editar RetenciónFactura	1	SI
5	Al modificar un campo de la retención total y luego presionar el ícono “Actualizar”	Muestra el mensaje: “ <i>La retención fue actualizada correctamente</i> ”	1	SI
6	Seleccionar retención y presionar el botón eliminar	Muestra un mensaje “ <i>La retención ha sido eliminada correctamente</i> ”	1	SI
7	Al presionar “Editar” un registro	Muestra el menú de operaciones para actualizar o eliminar si la fecha no es mayor a 15 días después de la fecha actual.	1	SI
Conclusión:	Las retenciones de totales solo se pueden editar con 15 días después de la fecha generada.			
Observación:	Las retenciones totales son las que se le enviarán a la presidencia.			

Tabla 4-22 Prueba de Caja Negra: Retenciones Totales, Datos Correctos
Fuente: El Autor, 2013

4.3.2 PRUEBAS DE CAJA BLANCA

Consiste en realizar pruebas para verificar que líneas específicas de código funcionan tal como está definido. También se le conoce como prueba de caja-transparente o pruebas estructurales.

La prueba de la caja blanca es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivar los casos de prueba.

Las pruebas de caja blanca intentan garantizar que:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo
- Se utilizan las decisiones en su parte verdadera y en su parte falsa
- Se ejecuten todos los bucles en sus límites
- Se utilizan todas las estructuras de datos internas.

Prueba No: 1	Validar Usuario y Contraseña
Propósito	Ingreso del sistema si el usuario y la contraseña son correctos
Código	<pre> public void validar() { List list_acceso = con.consultarHQL("from TbUsuario as us where us.idUsuario" + " = '" + cedula + "' and us.password = '" + clave + "'"); if (list_acceso != null && list_acceso.size() > 0) { try { current = (TbUsuario) list_acceso.get(0); usa = new BUsuario(current); FacesContext.getCurrentInstance().getExternalContext() .redirect("../index.xhtml"); FacesContext.getCurrentInstance().getExternalContext() .getSessionMap().put("nombre_usua", cedula); JsflUtil.addSuccessMessage(current.getNombres() + " " + current.getApellidos()); FacesMessage facesMsg = new FacesMessage(FacesMessage .SEVERITY_INFO, ResourceBundle.getBundle ("/resources/Bundle_es").getString("LoginCorrecto") , ResourceBundle.getBundle("/resources/Bundle_es") .getString("LoginCorrecto")); FacesContext.getCurrentInstance() .addMessage("successInfo", facesMsg); cargarMenu(current.getTbPerfil().getCodPerfil().toString()); logeado = true; } catch (Exception e) { System.out.println("Vino al catch"); } } else { JsflUtil.addErrorMessage(ResourceBundle .getBundle("/resources/Bundle_es").getString("LoginIncorrecto")); } } </pre>
Pre-requisitos	Iniciar el sistema ingresando un usuario existente en la base de datos.
Datos a utilizar	Usuario: useradmin1 Password: useradmin1*
Acciones	list_acceso = con.consultarHQL("from TbUsuario as us where us.idUsuario" + " = '" + cedula + "' and us.password = '" + clave + "'");
Resultado Final	Muestra la pantalla de inicio de sesión dependiendo el perfil del usuario con su respectivo menú, si no existe en la base de datos aparece un mensaje de error que indica que el usuario o el password es incorrecto.

Tabla 4-23 Prueba de Caja Blanca: Validar Usuario y Contraseña
Fuente: El Autor, 2013

Prueba No: 2	Establecer usuario de sesión
Propósito	Devolver la sesión que tiene el usuario ingresado.
Código	<pre> public BUsuario() { //Para saber si esta como administrador. if (getSession().getTbPerfil().getCodPerfil().equalsIgnoreCase("PER_01")) { admin = true; } else { admin = false; } /*PER_03 ingresa como asociado, para validar solo su sesion al ingresar*/ if (getSession().getTbPerfil().getCodPerfil().equalsIgnoreCase("PER_03")) { consultarPrestamoUser(); current = getSession(); } else { consultarUsuario(); current = new TbUsuario(); //Muestra el Usuario Logreado System.out.println("Sesion constructor: " + getSession().getTbPerfil().getCodPerfil()); } retencionUsuario(); } </pre>
Pre-requisitos	Iniciar el sistema ingresando un usuario existente en la base de datos.
Datos a utilizar	Usuario: useradmin1 Password: useradmin1*
Acciones	<pre> if (getSession().getTbPerfil().getCodPerfil().equalsIgnoreCase("PER_01")) { admin = true; } else { admin = false; } </pre>
Resultado Final	Establecer la sesión con el usuario que se haya ingresado.

Tabla 4-24 Prueba de Caja Blanca: Establecer Usuario de Sesión
Fuente: El Autor, 2013

Prueba No: 3	Consultar préstamos por usuario ingresado.
Propósito	Consultar los préstamos que tiene un asociado
Código	<pre> public void consultarPrestamoUser() { listaPrestamo = con.consultarHQL("from TbPrestamo u where " + "u.tbUsuario.idUsuario = '" + session.getIdUsuario() + "'"); } </pre>
Pre-requisitos	Ingresar como asociado y escoger la opción Préstamo
Datos a utilizar	Usuario: 0400429544 Password: 0400429544*
Acciones	

	listaPrestamo = con.consultarHQL("from TbPrestamo u where " + "u.tbUsuario.idUsuario = " + 0400429544 + "");
Resultado Final	Muestra todos los préstamos que ha realizado el usuario específico en el sistema

Tabla 4-25 Prueba de Caja Blanca: Consultar Préstamos por Usuario Ingresado
Fuente: El Autor, 2013

Prueba No: 4	Ingreso de Sueldo
Propósito	Ingresar el sueldo de un asociado debe ser mayor a 0.
Código	<pre>if (current.getTbPerfil().getCodPerfil().equals("PER_03")) { try { if (current.getSueldo().compareTo(cero) > 0); } catch (Exception e) { current.setSueldo(cero); } }</pre>
Pre-requisitos	Ingresar un nuevo usuario y dejar el campo sueldo vacío.
Datos a utilizar	Suedo: ""
Acciones	<pre>BigDecimal cero = new BigDecimal("0.00"); if (current.getSueldo().compareTo(cero) > 0); current.setSueldo(cero);</pre>
Resultado Final	El campo sueldo se establece como un valor BigDEcimal de 0.00

Tabla 4-26 Prueba de Caja Blanca: Ingreso de Sueldo
Fuente: El Autor, 2013

Prueba No: 5	Validar cédula
Propósito	Ingresar usuarios con cédulas reales
Código	<pre>public static boolean esCedula(String cedula) { if (!((cedula.length() == 10) && cedula.matches("[0-9]{10}\$"))) return false; } int prov = Integer.parseInt(cedula.substring(0, 2)); if (!((prov > 0) && (prov <= NUMERO_DE_PROVINCIAS))) { return false; } int[] d = new int[10]; for (int i = 0; i < d.length; i++) { d[i] = Integer.parseInt(cedula.charAt(i) + ""); } int imp = 0; int par = 0;</pre>

	<pre> for (int i = 0; i < d.length; i += 2) { d[i] = ((d[i] * 2) > 9) ? ((d[i] * 2) - 9) : (d[i] * 2); imp += d[i]; } for (int i = 1; i < (d.length - 1); i += 2) { par += d[i]; } int suma = imp + par; int d10 = Integer.parseInt(String.valueOf(suma + 10).substring(0, 1 "0")) - suma; d10 = (d10 == 10) ? 0 : d10; return d10 == d[9]; } </pre>
Pre-requisitos	Ingresar un nuevo usuario y seleccionar el perfil: Asociado
Datos a utilizar	cédula: 1718572041
Acciones	<pre> if (!((cedula.length() == 10) && cedula.matches("[0-9]{10}\$"))) { return false; } </pre>
Resultado Final	Retorna true, que corresponde a la cédula cumple con los dígitos validadores.

Tabla 4-27 Prueba de Caja Blanca: Validar Cédula
Fuente: El Autor, 2013

Prueba No: 6	Consultar cuota de préstamo
Propósito	Retornar el número de cuotas restantes que faltan cuando se crea un préstamo
Código	<pre> public void consultarCuota() { int i = current.getMesesAporte(); listCuota = new ArrayList(); for (int j = 0; j <= i; j++) { listCuota.add(j); } } </pre>
Pre-requisitos	Haber ingresado un préstamo o editar un préstamo
Datos a utilizar	MesesAporte: 4
Acciones	<pre> listCuota = new ArrayList(); for (int j = 0; j <= i; j++) { listCuota.add(j); } </pre>
Resultado Final	Devuelve en la lista los valores del 0 al 4

Tabla 4-28 Prueba de Caja Blanca: Consultar Cuota de Préstamo
Fuente: El Autor, 2013

Prueba No: 7	Ingresar código de clave primaria
Propósito	Buscar de la base de datos el último registro de campo primario y devolver aumentado en uno
Código	<pre> public String consultaCodigo() { try { List list_acceso = con.consultarHQL("from TbAportes" + "order by codAporte DESC"); cod = (TbAportes) list_acceso.get(0); String codigo = validador.ultimoCodigo(cod.getCodAporte().toString()); cod = new TbAportes(); return codigo; } catch (Exception e) { return "APO_01"; } } </pre>
Pre-requisitos	Presionar el ícono guardar en un aporte.
Datos a utilizar	APO_04
Acciones	cod = (TbAportes) list_acceso.get(0); String codigo = validador.ultimoCodigo(cod.getCodAporte().toString());
Resultado Final	Devuelve un string "APO_05"

Tabla 4-29 Prueba de Caja Blanca: Ingresar Código de Clave Primaria
Fuente: El Autor, 2013

Prueba No: 8	Guardar Aporte
Propósito	Guardar un aporte después de haber ingresado la información necesaria
Código	<pre> public void guardarAporte() { if (current.getCodAporte() == null) { current.setCodAporte(consultaCodigo()); } if (current.getMesesRestantes() == null) { current.setMesesRestantes(current.getMesesAporte()); } if (con.guardar(current) == true) { consultarAporte(); JsفUtil.addSuccessMessage(ResourceBundle. getBundle("resources/Bundle_es").getString("TbAportesCreated")); limpiar(); } else { JsفUtil.addErrorMessage(ResourceBundle. getBundle("resources/Bundle_es").getString("PersistenceErrorOccured")); } } </pre>
Pre-requisitos	Presionar el ícono guardar en un aporte.
Datos a utilizar	NombreAporte: ASOCIACION ASOCIADO Monto: 5 Nombramiento: SI

	Activo: 5 Detalle: APORTE MENSUAL Meses de Aporte: Indefinido
Acciones	if (current.getCodAporte() == null) { current.setCodAporte(consultaCodigo()); }
Resultado Final	El aporte fue creado exitosamente.

Tabla 4-30 Prueba de Caja Blanca: Guardar Aporte

Fuente: El Autor, 2013

Prueba No: 9	Modificar Retención de Aporte
Propósito	Al modificar una retención de aporte o eliminar retenciones de aportes
Código	<pre> public void modificarAporte(TbAportes aporte, int mes) { if (mes == 0) { aporte.setActivo("NO"); } aporte.setMesesRestantes(mes); if (con.modificar(aporte) == true) { consultarAporte(); //JsfUtil.addSuccessMessage(ResourceBundle.getBundle("resources/Bundle_es").getSt } else { JsفUtil.addErrorMessage(ResourceBundle .getBundle("resources/Bundle_es").getString("PersistenceErrorOccured")); } } </pre>
Pre-requisitos	Haber generado las retenciones de aportes para el mes correspondiente
Datos a utilizar	TbAporte: ASOCIACION ASOCIADO Mes: 1
Acciones	if (mes == 0) { aporte.setActivo("NO"); } aporte.setMesesRestantes(mes);
Resultado Final	El aporte fue eliminado o actualizado correctamente, Si los meses de aporte son igual a cero, el campo activo cambia a NO.

Tabla 4-31 Prueba de Caja Blanca: Modificar Retención de Aporte

Fuente: El Autor, 2013

Prueba No: 10	Cheque no relacionado
Propósito	No mostrar cheques que ya hayan sido utilizados
Código	<pre> public void chequeNoRelacionado() { listacheque = con.consultarHQL("from TbCheque " + "where codCheque not in (select " + "tbCheque.codCheque from TbPrestamo)" + " and codCheque not in (select " + "tbCheque.codCheque from TbFacturasXPagar)" + " order by codCheque DESC "); } </pre>
Pre-requisitos	Abrir una página con opción para agregar nueva factura o para agregar un nuevo préstamo
Datos utilizar a	
Acciones	<pre> if (mes == 0) { aporte.setActivo("NO"); } aporte.setMesesRestantes(mes); </pre>
Resultado Final	Me muestra en una lista del menú desplegable los cheques que no han sido utilizados, con esto se espera que el menú no sea demasiado grande para los usuarios y que no pueda escoger un cheque que ya ha sido otorgado para otra operación.

Tabla 4-32 Prueba de Caja Blanca: Cheque no Relacionado
Fuente: El Autor, 2013

Prueba No: 11	Facturas recientes
Propósito	Consultar facturas que no hayan sido creado 20 días antes de la fecha actual
Código	<pre> public Date generarFecha() { calendario.setTime(fechaDate); calendario.add(Calendar.DATE, -20); Date nuevaFecha2 = calendario.getTime(); System.out.println("La fecha es:... " + nuevaFecha2); return nuevaFecha2; } public void consultarFactura2() { Date fec = generarFecha(); lista2 = con.consultarHQL("from TbFacturasXPagar u " + "where u.fechaRecepcionFactura > '" + fec + "' " + "order by u.codFacPag DESC"); } </pre>
Pre-requisitos	Agregar una nueva retención de facturas.
Datos utilizar a	Seleccionar del menú desplegable una factura
Acciones	Date fec = generarFecha();

	lista2 = con.consultarHQL
Resultado Final	Me muestra en una lista del menú desplegable las facturas que tengan una fecha de creación de hasta 20 días antes de la fecha actual, con esto se espera que el menú no sea demasiado grande para los usuarios y que no pueda escoger una factura que haya sido para meses anteriores

Tabla 4-33 Prueba de Caja Blanca: Facturas Recientes

Fuente: El Autor, 2013

Prueba No: 12	Mostrar un Préstamo
Propósito	Detallar un préstamo.
Código	<pre> public void consultarCuota() { int i = current.getCuotas(); listCuota = new ArrayList(); for (int j = 0; j <= i; j++) { listCuota.add(j); } } public String preparaVer() { cuotas.setCuotas(current.getCuotas()); consultarCuota(); return "Edit"; } </pre>
Pre-requisitos	Haber creado un préstamo y dar clic en el ícono Editar de la lista de préstamos
Datos utilizar	a Asociado: Villamar Mora Karina NumCheque: 120454513 Cuotas: 3 CuotasRestantes:2
Acciones	cuotas.setCuotas(current.getCuotas()); consultarCuota();
Resultado Final	Muestra en una lista desplegable las cuotas restantes que tiene ese préstamo.

Tabla 4-34 Prueba de Caja Blanca: Mostrar un Préstamo

Fuente: El Autor, 2013

Prueba No: 13	Generar Retenciones de Aportes
Propósito	Generar las retenciones de los aportes que se encuentran activos y aplicarlos a los asociados que están activos.
Código	<pre> public void generarRetencionesAportes() { try { List list_fecha = con.consultarHQL("from TbRetencionAportes where " + "EXTRACT(YEAR FROM fechaRetApor) = " + "'" + year + "' AND EXTRACT(MONTH FROM fechaRetApor)='" + mes + "'"); if (list_fecha.size() > 0) { JsUtil.addErrorMessage(ResourceBundle.getBundle("resources/Bundle_es") .getString("RetencionesTotalesAportesIncorrectas") + " : " + fechaGeneral); } else { listaAporte = con.consultarHQL("from TbAportes where activo='SI' and " + "mesesAporte=13 or mesesRestantes > 0"); for (int i = 0; i < listaAporte.size(); i++) { listaAporte.get(i); //aporte = (TbAportes) list_aporte.get(i); String codigo = listaAporte.get(i).getCodAporte().toString(); String resultado = con.cargarSql("insert into tb_retencion_aportes " + "select u.id_usuario,a.cod_aporte,a.monto_retencion,now() " + "from tb_usuario u, tb_aportes a where cod_aporte=" + codigo + " " + "and u.cod_perfil='PER_03' and u.nombramiento = a.nombramiento " + "and a.activo = 'SI'"); if (listaAporte.get(i).getMesesAporte() < 13) { int menos = listaAporte.get(i).getMesesRestantes() - 1; Baporte aporteC = new Baporte(); aporteC.modificarAporte(listaAporte.get(i), menos); } } consultarFechasRet(); JsUtil.addSuccessMessage(ResourceBundle.getBundle("resources/Bundle_es") </pre>
Pre-requisitos	Haber creado aportes.
Datos utilizar	a Aportes creados y activos.
Acciones	Clic en el ícono generar retenciones de aportes, para el caso del administrador,
Resultado Final	Muestra un mensaje los aportes han sido creados exitosamente para el mes en curso.

Tabla 4-35 Prueba de Caja Blanca: Generar Retenciones de Aportes
Fuente: El Autor, 2013

Prueba No: 14	Eliminar Retenciones de Aportes
Propósito	Consultar las retenciones de aportes que han sido generadas 9 días antes de la fecha actual para poder eliminar.
Código	<pre> public void consultarFechasRet() { fechaElimar = null; calendario.setTime(fechaDate); calendario.add(Calendar.DATE, -9); Date nuevaFecha = calendario.getTime(); fechaGeneral2 = formatearFecha2(nuevaFecha); calendario.setTime(fechaDate); Date nuevaFechaActual = calendario.getTime(); String FechaGeneral3 = formatearFecha2(nuevaFechaActual); listafec = con.consultarSQL("select fecha_ret_apor from tb_retencion_aportes " + "where fecha_ret_apor between '" + fechaGeneral2 + "' " + "and '" + FechaGeneral3 + "' " + "group by fecha_ret_apor order by fecha_ret_apor desc"); } </pre>
Pre-requisitos	Haber generado retenciones de aportes en los últimos 9 días
Datos a utilizar	Retenciones generadas
Acciones	<pre> <h:commandLink action="/retAporte/Eliminar.xhtml" immediate="true" actionListener="#{bretAporte.consultarFechasRet()}"> <h:graphicImage value="../resources/css/images/eliminar41.png" width="35" title="Eliminar Retenciones de Aportes"/> </h:commandLink> </pre>
Resultado Final	En la pantalla de eliminar retenciones de aportes solo me aparece en una desplegable las retenciones que han sido generadas 9 días antes de la fecha actual.

Tabla 4-36 Prueba de Caja Blanca: Eliminar Retenciones de Aportes

Fuente: El Autor, 2013

Prueba No: 15	Eliminar Una Retención de Aportes
Propósito	Permitir o no mostrar los íconos para eliminar o actualizar una retención de aportes.
Código	<pre> public boolean EditEliminar() { calendario.setTime (fechaDate); calendario.add(Calendar.DATE, -9); Date nuevaFecha2 = calendario.getTime(); if (current.getFechaRetApor().compareTo(nuevaFecha2) > 0) { return true; } else { return false; } } </pre>
Pre-requisitos	Buscar retenciones de aportes, dar clic en una y presionar el ícono editar.
Datos a utilizar	MontoRetención: 10.00 FechaRetención: 02/Diciembre/2012 Asociado: Villamar Mora Karina Aporte: Asociación no asociado <i>FechaActual: 08/Enero/203</i>
Acciones	<pre> <c:if test="#{bretAporte.editElimi}"> <h:commandLink action="#{bretAporte.modificarRetAporte}" > <h:graphicImage value="../resources/css/images/actualizar.png" width="35" title="Actualizar Retención Aporte"/> </h:commandLink> <p:separator /> <h:commandLink action="#{bretAporte.eliminarRetAporte}" > <h:graphicImage value="../resources/css/images/eliminar.png" width="35" title="Eliminar Retención Aporte"/> </h:commandLink> </c:if> </pre>
Resultado Final	En el menú de acciones de la retención a editar, en este caso no se mostrará el ícono de eliminar ni el de actualizar, porque la fecha no es menor a 9 días de la fecha actual.

Tabla 4-37 Prueba de Caja Blanca: Eliminar una Retención de Aportes

Fuente: El Autor, 2013

Prueba No: 16	Editar o Eliminar Retenciones
Propósito	Permitir o no mostrar los íconos para eliminar o actualizar una retención de aportes.

Código	<pre> public boolean EditEliminar() { calendario.setTime (fechaDate); calendario.add(Calendar.DATE, -9); Date nuevaFecha2 = calendario.getTime(); if (current.getFechaRetApor().compareTo(nuevaFecha2) > 0) { return true; } else { return false; } } </pre>
Pre-requisitos	Buscar retenciones de aportes, dar clic en una y presionar el ícono editar.
Datos a utilizar	MontoRetención: 10.00 FechaRetención: 02/Diciembre/2012 Asociado: Villamar Mora Karina Aporte: Asociación no asociado <i>FechaActual: 08/Enero/203</i>
Acciones	<pre> <c:if test="#{bretAporte.editElimi}"> <h:commandLink action="#{bretAporte.modificarRetAporte}" > <h:graphicImage value="../resources/css/images/actualizar.png" width="35" title="Actualizar Retención Aporte"/> </h:commandLink> <p:separator /> <h:commandLink action="#{bretAporte.eliminarRetAporte}" > <h:graphicImage value="../resources/css/images/eliminar.png" width="35" title="Eliminar Retención Aporte"/> </h:commandLink> </c:if> </pre>
Resultado Final	En el menú de acciones de la retención a editar, en este caso no se mostrará el ícono de eliminar ni el de actualizar, porque la fecha no es menor a 9 días de la fecha actual.

Tabla 4-38 Prueba de Caja Blanca: Editar o Eliminar Retenciones

Fuente: El Autor, 2013

Prueba No: 17	Generar Retenciones de Préstamos
Propósito	Generar las retenciones de préstamos que se encuentran activos y aplicarlos a los asociados que están activos, cuyas cuotas restantes sean mayores a cero.
Código	<pre> public void generarRetencionPrestamo() { Double nombramiento2; try { List list_fecha = con.consultarHQL("from TbRetencionPrestamo" + " where EXTRACT(YEAR FROM fechaRetPres) = " + "'" + year + "' AND EXTRACT(MONTH FROM" + " fechaRetPres)='" + mes + "'"); if (list_fecha.size() > 0) { JsUtil.addErrorMessage(ResourceBundle .getBundle("resources/Bundle_es") .getString("RetencionesTotalesPresIncorrectas") + " : " + fechaGeneral); } else { listaPrestamo = con.consultarHQL("from TbPrestamo where " + "cuotasRestantes > 0 and fechaPrestamo < now()"); for (int i = 0; i < listaPrestamo.size(); i++) { listaCheque = con.consultarHQL("from TbCheque where codCheque " + "= '" + listaPrestamo.get(i).getTbCheque() + ".getCodCheque() + "'"); current = new TbRetencionPrestamo(); current.setTbUsuario(listaPrestamo.get(i).getTbUsuario()); System.out.println(current.getTbUsuario().getIdUsuario()); current.setTbPrestamo(listaPrestamo.get(i)); System.out.println(current.getTbPrestamo().getCodPrestamo()); listUsuario = con.consultarHQL("from TbUsuario where idUsuario " + "= '" + listaPrestamo.get(i) + ".getTbUsuario().getIdUsuario() + "' "); if (listUsuario.get(0).getNombramiento().equals("SI")) { nombramiento2 = 0.02; } } } } catch (Exception e) { e.printStackTrace(); } } </pre>
Pre-requisitos	Haber creado préstamos.
Datos a utilizar	Préstamos activos con cuotas restantes mayores a cero.
Acciones	Clic en el ícono generar retenciones de préstamos, para el caso del administrador.
Resultado Final	Muestra un mensaje los préstamos han sido creados exitosamente para el mes en curso.

Tabla 4-39 Prueba de Caja Blanca: Generar Retenciones de Préstamos
Fuente: El Autor, 2013

Prueba No: 18	Generar Retenciones Totales
Propósito	Generar las retenciones totales aplicarlos a los asociados que están activos.
Código	<pre> public void generarRetenctionesTotal() { //mes = 9; if (fechaHasta.compareTo(fechaDesde) > 0) { try { List list_VerificaFechas = con.consultarHQL("from " + "TbRetencionTotal where '" + fechaDesde + "' " + "between fechaDesde and fechaHasta or " + "'" + fechaHasta + "' " + "between fechaDesde and fechaHasta"); if (list_VerificaFechas.size() > 0) { dialogo = "null"; JsفUtil.addErrorMessage(ResourceBundle .getBundle("resources/Bundle_es") .getString("RetencionesTotalesFacIncorrectas") + " : " + formatearFecha(fechaHasta)); fechasVacias(); } else { fechaGeneral4 = obtenerFechaActual(); con.cargarSql("delete from tempR"); con.cargarSql("insert into tempR " + "select sum(a.monto_ret_pres), u.id_usuario " + "from tb_retencion_prestamo a,tb_usuario u " + "where u.id_usuario = a.id_usuario " + "and fecha_ret_pres between '" + fechaDesde + "' and '" + fechaHasta + "' " + "group by u.id_usuario;"); con.cargarSql("insert into tb_retencion_total " + "select u.id_usuario,sum(sum),' + fechaDesde + "','" + fechaHasta + "','" + fechaGeneral4 + "' " + "from tempR a,tb_usuario u " + "where u.id_usuario = a.id_usuario group by u.id_usuario;"); consultarRetTotal(); List list_total = con.consultarHQL("from TbRetencionTotal " + "where fechaDesde = '" + fechaDesde + "' " + "and fechaHasta = '" + fechaHasta + "'"); registros = list_total.size(); JsفUtil.addSuccessMessage(ResourceBundle.getBundle("resources/Bundle_es") .getString("RetencionesTotalesFacCorrectas") + " : " + fechaGeneral4); dialogo = "dialog.show()"; consultarFechasRetTotal(); fechasVacias(); } } catch (Exception e) { dialogo = "Error al generar retenciones totales: " + e.getMessage(); } } } </pre>
Pre-requisitos	Haber generado como administrador retenciones de aportes, retenciones de préstamos y retenciones de facturas.
Datos a utilizar	Retenciones de aportes, retenciones de préstamos y retenciones de facturas, con un mismo mes de generación.
Acciones	Clic en el ícono generar retenciones totales, para el caso del administrador.
Resultado Final	Muestra un mensaje las retenciones totales fueron generadas exitosamente.

Tabla 4-40 Prueba de Caja Blanca: Generar Retenciones Totales
Fuente: El Autor, 2013

Prueba No: 19	Generar Todas las Retenciones
Propósito	Generar todas las retenciones y aplicarlos a los asociados que están activos para el mes en curso.
Código	<pre> public void generarTodasRetenciones() { if (fechaHasta.compareTo(fechaDesde) > 0) { try { BretAporte retAporte = new BretAporte(); retAporte.generarRetenccionesAportes(); BretPrestamo retPrestamo = new BretPrestamo(); retPrestamo.generarRetenccionPrestamo(); generarRetenccionesTotal(); } catch (Exception e) { JsفUtil.addErrorMessage(ResourceBundle .getBundle("resources/Bundle_es") .getString("PersistenceErrorOccured")); fechasVacias(); } } else { fechasVacias(); dialogo = "null"; JsفUtil.addErrorMessage(ResourceBundle .getBundle("resources/Bundle_es").getString("FechasNoValida")); } } </pre>
Pre-requisitos	Ingresar al sistema como secretaria después de tener préstamos y aportes activos al igual que haber ingresado facturas de las casas comerciales.
Datos utilizar	a Retenciones de aportes, retenciones de préstamos y retenciones de facturas, con un mismo mes de generación.
Acciones	Clic en el ícono presionar el ícono generar retenciones para el mes en curso,, para el caso del administrador.
Resultado Final	Muestra un mensaje las retenciones fueron generadas exitosamente.

Tabla 4-41 Prueba de Caja Blanca: Generar Todas las Retenciones

Fuente: El Autor, 2013

Prueba No: 20	Generar Retenciones ya generadas
Propósito	No permitir generar retenciones que ya hayan sido generadas para el mes en curso.
Código	<pre> public void generarRetenccionesTotal() { //mes = 9; if (fechaHasta.compareTo(fechaDesde) > 0) { try { List list_VerificaFechas = con.consultarHQL("from " + "TbRetencionTotal where '" + fechaDesde + "' " + "between fechaDesde and fechaHasta or " + "' '" + fechaHasta + "' " + "between fechaDesde and fechaHasta"); if (list_VerificaFechas.size() > 0) { dialogo = "null"; JsUtil.addErrorMessage(ResourceBundle .getBundle("resources/Bundle_es") .getString("RetencionesTotalesFacIncorrectas") + " : " + formatearFecha(fechaHasta)); fechasVacias(); } else { </pre>
Pre-requisitos	Haber generado retenciones para cualquier caso: retenciones de aportes, retenciones de préstamos, retenciones de facturas, retenciones totales.
Datos a utilizar	Retenciones de aportes, retenciones de préstamos y retenciones de facturas, con un mismo mes de generación.
Acciones	Clic en el ícono presionar el ícono generar retenciones para el mes en curso.
Resultado Final	Muestra un mensaje de error las retenciones ya han sido generadas para el mes en curso fecha actual: <i>fechaactual</i> .

Tabla 4-42 Prueba de Caja Blanca: Generar Retenciones ya Generadas
Fuente: El Autor, 2013

CAPITULO V

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se ha logrado conocer la situación actual de la Asociación de la Presidencia de la República del Ecuador, obteniendo información valiosa para el desarrollo del proyecto, se determinó que la forma más adecuada para estructurar los procesos de automatización y control tanto para el registro de información como la generación de retenciones de ingresos y egresos es mediante el uso de la metodología XP.
- La metodología XP, se adecuó perfectamente a los cambios suscitados a lo largo del proyecto de titulación, consiguiendo efectuar cada una de las etapas, con el propósito de simplificar el trabajo y optimizar los requerimientos del usuario, mediante un desarrollo ordenado de la delegación de las tareas en las historias de usuarios.
- La aplicación fue desarrollada con herramientas libres para Java como JSF y Primefaces, es así que se convierte en un software multiplataforma que no requiere de costos por licencias, adecuándose al mandato presidencial.- A través del análisis en el desarrollo se ha podido lograr una interfaz amigable para los usuarios del sistema con pantallas que no requieren de un mayor esfuerzo para efectuar los procesos que en un inicio se manejaba manualmente, de esta manera es posible controlar los ingresos y egresos mensuales, de acuerdo a sus proveedores, préstamos emitidos y aportes propios de la asociación de forma sencilla.

- La herramienta PostgreSQL 9.2.1 contribuyó con su potente base de datos para manejar la información en el momento de guardar y devolver parámetros necesarios, acoplándose extraordinariamente con la tecnología Hibernate, brindando confiabilidad en el traspaso de datos.
- Los requerimientos iniciales del software fueron completados, con una aplicación móvil, la cual permite a los socios acceder a la información detallada de sus retenciones generadas mensualmente.
- El servicio de mensajería que se implementó permite a los asociados estar al tanto de noticias generales y un aviso mensual con el detalle personal de sus retenciones.
- Un factor importante en la comunicación de los usuarios del sistema, al permitir la interrelación con cada uno de sus miembros, se lo realiza mediante sesiones de chat y envíos de correo electrónico, que permite compartir información a través de texto sobre algún tema de interés general.
- El uso de Web Service ayudó significativamente a la interconexión que se da en ambientes Web, al enviar y recibir información. Permitiendo de esta manera enviar las retenciones mensuales que automatizan el sistema desarrollado (AsoPresi) y de la misma manera, recibir información referente a las retenciones de los asociados que se generan en la Presidencia.
- En lo referente a la asistencia en el otorgamiento de préstamos, se implementó y automatizó un proceso que permite conocer la capacidad de endeudamiento que tiene el asociado interesado en realizar un préstamo, verificando los parámetros definidos por la asociación antes de ser aprobado, como son el gasto enviado por la presidencia, pertenecer a la asociación por más de un mes, verificando previamente, que la diferencia del sueldo percibido con respecto a estos parámetros no sea superior al 70% según la ley LOSEP.

5.2 RECOMENDACIONES

- Para lograr los beneficios esperados en el uso de la herramienta desarrollada, se recomienda que el personal administrativo y los socios deben capacitarse por medio de charlas y prácticas de uso del sistema, con la guía del manual de usuario, garantizando la efectividad en las respectivas tareas que realizan los diferentes módulos del sistema.
- Debido al constante cambio de las versiones de las herramientas de navegación, se recomienda probar la aplicación en las últimas versiones del navegador Firefox, para evitar posibles problemas de incompatibilidad con el sistema AsoPresi.
- En vista que los datos ingresados y la información generada automáticamente se almacena en una base de datos, se sugiere realizar respaldos mensuales de la misma, mediante la utilización de servidores de backups que realizarían de forma automática este proceso.
- Cargar la información referente a las retenciones que genera la Presidencia para contar con datos actualizados al momento de verificar la capacidad de endeudamiento de un socio.
- Al contar con un sistema para la automatización, manejo control de la información de la asociación se recomienda que el sistema sea utilizado al máximo y así se garantizará estabilidad y por ende confianza en la ejecución de los procesos.
- Los asociados deben mantener actualizada su información referente al número de celular que usan ya que esa es una de las maneras para recibir la

información, además si cuentan con dispositivos Android deberán contar con el servicio de internet para usar la aplicación suministrada por la asociación.

- Es fundamental tener en cuenta que ninguna aplicación ni sistema tiene un porcentaje del 100% con respecto a la seguridad. Para evitar problemas se recomienda el uso del servicio https para páginas Web y conexiones seguras con la autenticación de usuario y contraseña para el Web Service.
- Debido a los constantes requerimientos que se van produciendo en el software es ineludible contar con un mayor número de reportes, que se adecuen a los cambios de información.

GLOSARIO DE TÉRMINOS

Este Glosario está basado en los términos que se han utilizado a lo largo del desarrollo de este documento; además, se ha llevado a cabo un esfuerzo adicional en cuanto a la coherencia y el afinamiento de algunos términos que se usan en la actualidad para su respectiva comprensión y entendimiento. Las definiciones se describen de forma independientes en el glosario.

A

Android: Sistema operativo basado en el núcleo Linux diseñado originalmente para dispositivos móviles

AsoPresi: Nombre determinado para el sistema de la Asociación de Empleado de la Presidencia de la República del Ecuador

B

Bean: Llamado JavaBean o bean, es un componente hecho en software que se puede reutilizar, evitando programar cada uno de los componentes y que puede ser manipulado visualmente por una herramienta de programación en lenguaje Java.

Black box system tests: pruebas de caja negra que se realiza sobre el sistema

Builds: reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible.

C

Centos: sistema operativo que se encuentra basada en el código fuente de Red Hat Enterprise.

D

Debian: comunidad conformada por desarrolladores y usuarios, que mantiene un sistema operativo GNU basado en software libre

Dispositivos móviles: computadoras móviles pequeñas que entran en un bolsillo.

E

EJB's: Enterprise Java Bean, encapsula los objetos y se los referencia especialmente cuando se ocupa persistencias.

Entity: Es una clase entidad que son POJO's que representa una tabla de la base de datos, y cada instancia de esta clase representa un registro de la tabla, se consigue crear un mapeo entre una clase y una tabla.

F

Fine tuning: Tareas de ajuste entre cada fase de la metodología xp.

Framework: Estructura conceptual y tecnología de soporte definido, sirve para tener facilidad en el desarrollo y más organizado, representa una arquitectura de software y provee una estructura en la metodología de trabajo.

FTP: Es un servicio que los usuarios lo utilizan frecuentemente cuando se descargan desde la internet: drivers, música, documentos, etc.

G

GlassFish: Servidor de aplicaciones desarrollado por Sun Microsystems que implementa las tecnologías definidas en la plataforma Java.

GLP: Licencia Pública General GNU,

H

Hibernate: Es una herramienta para la plataforma Java que facilita el mapeo de los atributos entre una base de datos relacional y el modelo de objetos de aplicación mediante archivos declarativos XML.

I

Iteration Plan: Plan de iteraciones que se genera en cada fase de la metodología XP.

INEN: Organismo oficial de la República del Ecuador para la normalización, la certificación y la metrología.

L

LOSEP: Ley Orgánica del Servicio Público, regula el servicio público, a fin de contar con normas que respondan a las necesidades del recurso humano que labora en las instituciones y organismos del sector público.

J

Jasper: Se encarga de compilar las clases para convertirlos en servlets

Java: Lenguaje de programación orientado a objetos.

Jboss: Es un servidor de aplicaciones J2EE de código abierto implementado para asignar permisos de acceso a portales basados en la definición de roles.

JEE: Es la primera edición empresarial del paquete Java creada y distribuida por Sun Microsystems. Comprende un conjunto de especificaciones y funcionalidades orientadas al desarrollo de aplicaciones empresariales.

J2EE: Es la segunda edición empresarial del paquete Java creada y distribuida por Sun Microsystems. Comprende un conjunto de especificaciones y funcionalidades orientadas al desarrollo de aplicaciones empresariales.

JPA: Java Persistence API es el estándar de Java para persistencia.

Join: Permite combinar registros de dos o más tablas en una base de datos relacional.

JSF: (Java Server Faces), Tecnología y framework para aplicaciones java en ambiente Web que reduce el desarrollo de interfaces de usuario.

JUnit: Ambiente de prueba mediante librerías que ayuda a realizar pruebas de otras clases.

N

NTFS: Sistema de archivos diseñado específicamente para Windows NT, y utilizado por las versiones recientes del sistema operativo Windows.

P

Pair-programming: Programación en pareja en cada fase de la metodología XP.

Planning workshop: Planeamiento de trabajo en grupo para desarrollo de código de la metodología XP.

POJO: (Plain Old Java Object) enfatizar el uso de clases simples y no dependen de un framework en especial.

Postgresql: Sistema de gestión de bases de datos objeto-relacional.

Private: Declaración de variables, tipo de datos, métodos, objetos que solo puede ser usada en la clase en la que fueron creados.

Primefaces: Es un componente para Java Server Faces (JSF), de código abierto que cuenta con un conjunto de componentes que facilitan el diseño de la interfaz Web.

Protected: Declaración de variables, tipo de datos, objetos métodos que solo puede ser usado en la clase en la que fueron creados.

Public: Declaración de variables, tipo de datos, métodos, objetos que solo puede acceder desde fuera de la clase, sin necesidad de una instancia de la clase

R

Refactoring: Recodificación del código en una fase de la metodología XP.

Release Plan: Plan de entrega de cada iteración en inicio de una fase en la metodología XP.

RUP: Metodología que ordenar y estructurar el desarrollo de software y transformar los requisitos del usuario en un sistema Software.

S

Servlets: Pequeño programa que corre en un servidor, por lo general son aplicaciones Java que corren en un entorno de servidor Web.

Scrum: Proceso mediante el cual se aplica de manera regular un conjunto de mejores prácticas para trabajar en equipo y obtener los mejores resultados.

SMS: Short Message Service, servicio de mensajes cortos, es un servicio disponible en teléfonos móviles que permite enviar de mensajes de texto.

Spikes: Pequeños programas de prueba que evalúa el código de desarrollo.

Sqlite: Proyecto de dominio público que implementa una pequeña librería de aproximadamente 500kb, programado en el lenguaje C que tiene con función hacer de un sistemas de bases de datos relacional.

Smartphone: Es un teléfono inteligente que puede comunicarse a través de Wi-Fi, bluetooth, conexión a internet, envío de mensajería, e-mails.

Sun: Permite al usuario hacer más con menos, manejar elevadas cargas de tráfico y procesar grandes volúmenes de datos.

T

Tablets: Periférico o dispositivo digital portátil con las prestaciones propias de un PC.

Test-driven programming: Programación dirigida por las pruebas en las fases de la metodología Xp.

Tomcat: Es un contenedor de servlets que se utiliza en la Referencia oficial de la implementación para Java Servlet.

U

UML: Lenguaje para modelamiento unificado, es un lenguaje para la especificación, visualización, construcción y documentación de los artefactos de un proceso de sistema intensivo.

Unix: Es un sistema operativo portable, multitarea y multiusuario.

W

WebMail: Servicio que permite gestionar el correo electrónico desde un sitio Web, sin descargar los mensajes al propio ordenador.

Web Service: Es un componente software que puede ser registrado, descubierto e invocado mediante protocolos estándares de Internet.

X

XP: Método que más popularidad ha alcanzado de las metodologías ágiles

Se basa en la suposición de que es posible desarrollar a pesar del cambio continuo

BIBLIOGRAFÍA

- [1] Kent Beck, *Una explicación de la Programación extrema: aceptar el cambio.*: Agapea, 2002. [Online]. www.xprogramming.com
- [2] Oiver Andres Perez A. (2011, Junio) Cuatro enfoques metodológicos para el desarrollo de Software RUP – MSF – XP -SCRUM.
- [3] Jose Joskowicz, *Reglas y Prácticas en eXtreme Programming.*, 10/02/2008.
- [4] Patricio Salinas Caro. www.dcc.uchile.cl [Online]. <http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>
- [5] informatica.gob.ec. [Online]. <http://www.informatica.gob.ec/index.php/titulares-historico/sistemas-informaticos-historico-titulares/182-sistema-de-compras-publicas-realizado-con-software-libre>
- [6] wiki.centos.org. [Online]. <http://wiki.centos.org/es/Documentation>
- [7] F. BANCILHON, D. J. DEWITT, K. R. DITTRICH, D. M. P. ATKINSON,,: SIGMOD Conference, May 1990.
- [8] docs.jboss.org. [Online]. http://docs.jboss.org/hibernate/orm/3.5/reference/es-ES/html_single/
- [9] www.jtech.ua.es. [Online]. <http://www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/sa/sesion1-apuntes.htm>
- [10] glassfish.java.net. [Online]. <http://glassfish.java.net/es/>
- [11] <http://www.jboss.org/>. [Online]. www.jboss.org
- [12] ssbea.mercer.edu. [Online]. http://ssbea.mercer.edu/hogan/postgres_dbms.htm
- [13] Mario Saffirio. msaffirio. [Online]. <http://msaffirio.wordpress.com/2006/02/05/%C2%BFque-son-los-web-services/>
- [14] [Online]. <http://es.wikipedia.org/wiki/Android>
- [15] derechoecuador. [Online]. http://www.derechoecuador.com/index.php?option=com_content&view=article&id=6253:registro-oficial-no-477-viernes-24-de-junio-de-2011&catid=349:junio&Itemid=604
- [16] centos. [Online]. <http://www.centos.org/>
- [17] kioskea. langages. [Online]. <http://es.kioskea.net/contents/langages/langages.php3>
- [18] BERTRAND MEYER,,: Prentice Hall, 1997.
- [19] Apache Tomcat Project. tomcat.apache.org. [Online]. <http://tomcat.apache.org/>
- [20] technet.microsoft.com. technet.microsoft.com. [Online]. <http://technet.microsoft.com/es-ES/windowsserver/default.aspx>
- [21] ubuntu.com. [Online]. <http://www.ubuntu.com/business/server>
- [22] Paola Andrea Vargas. [Online]. <http://www.conexioncentral.com/blog/2012/12/13/sistemas-operativos-android-vs-ios/>
- [23] arachnoid.com. [Online]. <http://arachnoid.com/MySQL/>
- [24] Edwin Aguiar. elobservadordelsur. [Online]. <http://es.scribd.com/doc/7253619/Sqlite-Introduccion>

ANEXOS

ANEXO A. CAPACIDAD DE ENDEUDAMIENTO SEGÚN LOSEP

Considerando:

Que, por resolución de la Asamblea Nacional, se expide la Ley Orgánica del Servicio Público (LOSEP), publicada en el Suplemento del Registro Oficial No. 294 de 6 de octubre del 2010;

Que, mediante Decreto Ejecutivo No. 710, publicado en el (Registro Oficial No. 418 de 1 de abril del 2011) R. O. (SP) abril 01 No. 418 de 2011, se expide el Reglamento General a la Ley de Servicio Público, cuyo artículo 255 regula el procedimiento para la concesión del anticipo de remuneraciones;

Que, mediante Acuerdo Ministerial No. 00054 de 11 de marzo del 2011, publicado en el Registro Oficial No. 404 del 15 de marzo del 2011, el Ministerio de Relaciones Laborales, expidió el Reglamento y procedimiento para la concesión de anticipos, a que hace referencia el último inciso del artículo 255 del reglamento general ibídem; y,

En ejercicio de las facultades determinadas en el literal a) del artículo 4 del Acuerdo Ministerial No. 20110009,

Acuerda:

Expedir el Reglamento de anticipos de remuneraciones para el personal del Ministerio de Turismo.

Art. 1.- Tendrán derecho a solicitar anticipos la o el servidor público, que se encuentren prestando servicios en el Ministerio de Turismo, bajo la modalidad de nombramiento, contrato de servicios ocasionales y contratos de trabajo a plazos fijo e indefinido.

De igual forma, los servidores de otras instituciones que se encuentren declarados en comisión de servicios con remuneración y que perciban diferencias de remuneración tendrán derecho a solicitar anticipo. Para calcular el monto máximo de estos anticipos, se tomará en cuenta únicamente el valor de la diferencia de remuneración mensual unificada cancelada por el Ministerio.

DE LOS TIPOS DE ANTICIPOS

Art. 2.- Con cargo a las remuneraciones mensuales unificadas señaladas en el presupuesto institucional, las unidades responsables de la gestión financiera, a pedido de la o el servidor de la institución, sin necesidad de justificación previa podrán conceder uno de los siguientes anticipos:

Una remuneración mensual unificada del servidor, este anticipo concedido, se descontará hasta en un plazo de 60 días de otorgado el anticipo al momento de efectuar el pago de remuneraciones, de acuerdo con la programación de pagos aprobada por la Dirección de Administración de Talento Humano, o de ser el caso cuando la o el servidor público cesare en funciones antes del plazo concedido.

Hasta tres remuneraciones mensuales unificadas, este anticipo concedido será recaudado al momento de realizar el pago de las remuneraciones, de acuerdo con la programación de pagos aprobada por la Dirección de Administración de Talento Humano, dentro del plazo solicitado por la o el servidor, que no excederá los 12 meses, contados desde la concesión del anticipo en el caso de las y los servidores públicos con nombramiento, y del tiempo estipulado contractualmente para el caso de servidores públicos con contrato de servicios ocasionales. El descuento por el anticipo concedido, se efectuará mensualmente, de manera prorrateada durante el plazo convenido; excepto en el mes de diciembre, en el cual el descuento corresponderá por lo menos al 70% del valor de una remuneración mensual unificada de la o el servidor, que será descontado del décimo tercer sueldo.

Art. 3.- Previo a la entrega del anticipo de remuneración, la o el servidor autorizará expresamente el débito periódico del valor del anticipo en el pago de sus haberes, mediante el formulario elaborado para la concesión del anticipo; y, en caso de cesación de funciones o terminación del contrato de servicios ocasionales, autorizará se le descuente de su liquidación de haberes, íntegra y totalmente los valores y montos a que hubiese lugar.

REQUISITOS Y CONDICIONES

Art. 4.- Para la concesión de los anticipos se observarán las siguientes disposiciones:

Los anticipos se concederán a las o los servidores con nombramiento permanente, funcionarios de libre nombramiento y remoción, contratos fijos, contratos bajo el Código del Trabajo y contrato de servicios ocasionales, para estos tres últimos, la programación del descuento no podrá superar la fecha de terminación del contrato, en el caso de los funcionarios de libre nombramiento se concederá previo el análisis de su tiempo de servicio y los valores de sus liquidaciones en el caso de concluir sus funciones;

Las o los servidores que fueren declarados en comisión de servicio sin remuneración, deberán cancelar inmediatamente el saldo que tuvieran pendiente por concepto de anticipos. La Dirección de Administración de Talento Humano, descontará los saldos adeudados por anticipos de la última remuneración percibida y de los haberes a los que tenga derecho la o el servidor. De mantener compromisos pendientes, deberá realizar depósitos a la cuenta de ingresos propios de la institución en un plazo máximo de 30 días;

No se podrá acceder a dos anticipos en forma simultánea, tampoco podrán proceder a la renovación de los anticipos otorgados;

No se concederán anticipos en el mes de diciembre;

Se podrá pre cancelar con fondos propios los anticipos que se les hubiere otorgado, pero no podrán solicitar uno para cancelar otro anticipo vigente. Dicha pre cancelación se realizará directamente a la cuenta de ingresos que mantiene la institución;

Para la concesión de los anticipos la Dirección de Administración de Talento Humano, verificará con la documentación de cada servidora o servidor la capacidad de pago del solicitante;

Vfb hgbgn

Los anticipos se otorgarán en función de las disponibilidades financieras con cargo a las remuneraciones mensuales unificadas u honorarios señalados en el vigente presupuesto del Ministerio de Turismo y dependiendo de la capacidad de endeudamiento del servidor.

La Dirección de Administración de Talento Humano, establecerá el límite de endeudamiento de cada solicitante, considerando para ello que la o el servidor perciba una remuneración líquida mensual no menor al 20% de la respectiva remuneración mensual unificada, luego de deducidos los descuentos del anticipo solicitado y demás retenciones mensuales calculadas sobre la base del mes anterior; y,

Los anticipos se entregarán mediante transferencia a nombre de la o el servidor solicitante. [15]

ANEXO B. MANUAL DE USUARIO

1 INTRODUCCIÓN.

El motivo fundamental de este documento es el esclarecimiento de las operaciones realizables con la parte ejecutable del proyecto, la cual se basa, en el manejo del sistema de la Asociación de la Presidencia de la República del Ecuador.

1.1 DESCRIPCIÓN DEL PROYECTO

El sistema para la Asociación de la Presidencia de la República del Ecuador pretende agilizar el proceso de atención a los asociados tanto en la consulta como en el mecanismo de retención al reducir los papeles, formularios y trámites tanto para el asociado como la secretaria y el administrador.

El presente manual tiene por objetivo instruir al usuario en el uso operativo del sistema para la Asociación de la Presidencia de la República del Ecuador en los cuanto a conceptos de configuración, administración y operación.

2 INGRESO AL SISTEMA

Para ingresar al sistema, ingresar al portal web de la asociación de la presidencia de la República del Ecuador. Digite su nombre de usuario y clave (Figura 1).



Figura 1 Ingreso del Portal Web
Fuente: El Autor, 2013

Si el usuario o la clave no son las correctas le aparecerá un mensaje de advertencia que indica que el usuario o la contraseña son erróneos (Figura 2).



Figura 2 Mensaje de Advertencia de Inicio de Sesión
Fuente: El Autor, 2013

Si el usuario y la contraseña son correctos se mostrará una pantalla (Figura 3) con el menú principal que contiene:

- Descripción del perfil que se ha ingresado.
- Menú principal.
- Información general de la asociación
- Botones de ingreso y salida de la aplicación



Figura 3Página Principal del Usuario Logeado

Fuente: El Autor, 2013

2.1 MÓDULOS DEL SISTEMA

Los requerimientos se agrupan y forman los módulos del sistema detallados a continuación.

3 MÓDULO ADMINISTRADOR

El administrador Ingresa al sistema como súper usuario, el cual podrá tener información resumida de la información que se lleva en la asociación como:

- Creación y administración de usuarios y perfiles.
- Registro de bancos y cuentas bancarias.

3.1 MENÚ USUARIO

El menú Usuario tiene como opción la creación de usuarios. Para esto, haciendo clic en el menú, se despliega una barra de tareas en la cual se encuentran los botones que se muestran como imágenes, tales como:

- Nuevo Usuario
- Home
- Salir

Debajo de la barra de tareas se encuentra la lista de usuarios que se despliega en una grilla. En la última columna se encuentra un botón llamado Editar junto con una imagen que ilustra la funcionalidad del botón (Figura 4).

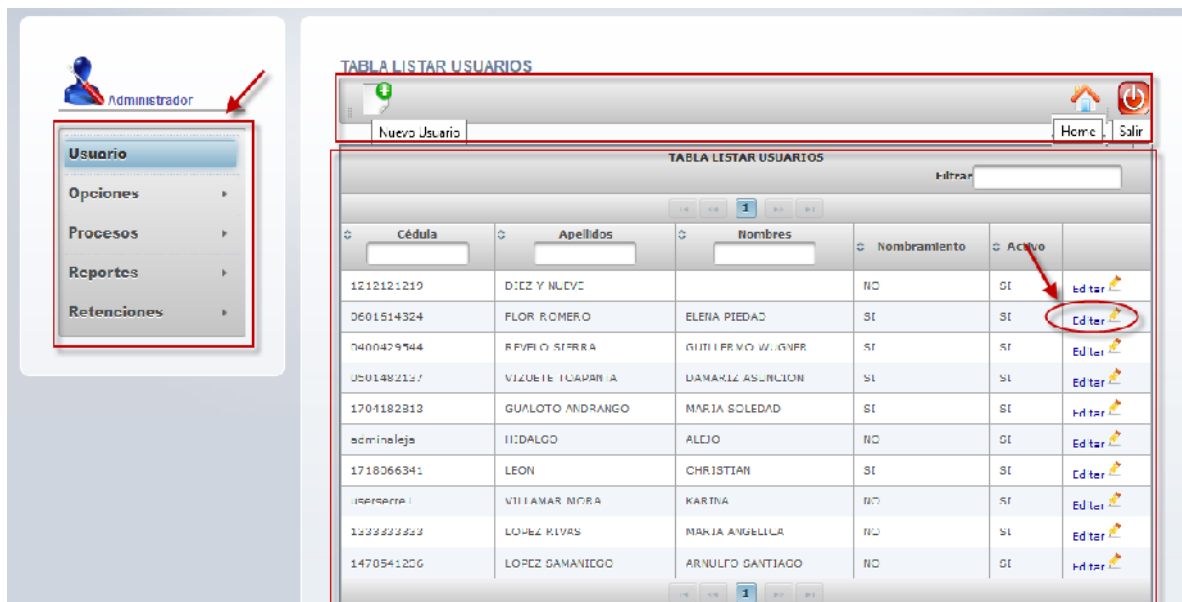


Figura 4Lista de Usuarios

Fuente: El Autor, 2013

Haciendo clic en el botón Nuevo Usuario se despliega una grilla para con la información del usuario para editar.

⚠ Importante.

El administrador no puede editar la cédula y contraseña.

Arriba de la grilla se encuentra una barra de tareas (Figura 5) donde se muestran los botones:

- Nuevo usuario
- Actualizar usuario
- Eliminar usuario
- Restablecer contraseña

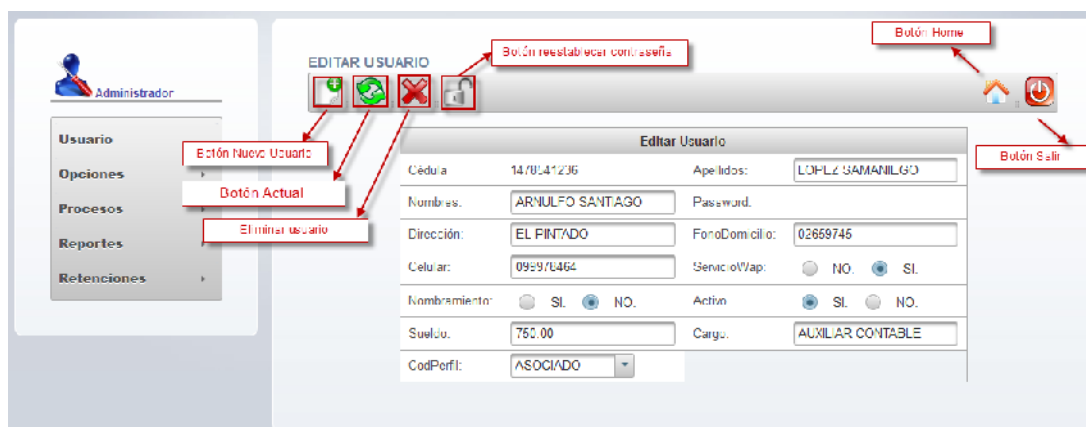


Figura 5 Editar Información del Usuario
Fuente: El Autor, 2013

Después de editar la información que se requiere. Haciendo clic en el botón Actualizar usuario se despliega un mensaje de información que se ha registrado el usuario (Figura 6).



Figura 6 Mensaje de Información de Actualización de Datos
Fuente: El Autor, 2013

Si se desea generar de nuevo la contraseña, haciendo clic en el botón restablecer contraseña se ha fijado una nueva contraseña para el usuario, desplegándose un de advertencia (Figura 7).



Figura 7 Mensaje de Advertencia Contraseña Restablecida
Fuente: El Autor, 2013

Importante.

El administrador puede restablecer la contraseña solo con la autorización del asociado.

En el caso de que el asociado ya no pertenezca a la asociación, haciendo clic en el botón eliminar se excluye al asociado del sistema. Se despliega un mensaje de información sobre la eliminación exitosa (Figura 8).



Figura 8 Mensaje de Advertencia Eliminación Asociado
Fuente: El Autor, 2013

Como última acción de este menú es la agregación de asociados. Haciendo clic en el botón Nuevo usuario, se despliega una grilla para ingresar la nueva información del usuario como:

- Cédula : identificación del asociado, campo obligatorio
- Nombres: nombres del asociado, campo requerido
- Dirección: dirección domiciliaria del asociado.
- Celular: teléfono celular
- Nombramiento: el asociado puede o no tener nombramiento dentro de la presidencia de la república del Ecuador.
- Sueldo: ingreso del valor salarial del asociado
- Código de perfil: tipo de perfil que posee el asociado.
- Apellidos: apellidos del asociado, campo requerido.
- Contraseña: contraseña del asociado.
- Servicio WAP: registro de servicio Wap si posee el asociado
- Fono Domiciliario: teléfono domiciliario
- Activo: miembro activo o pasivo de la asociación
- Cargo: descripción del cargo del asociado.

Arriba de la grilla se encuentra una barra de tareas (Figura 9) con los siguientes botones:

- Nuevo usuario
- Guardar usuario
- Limpiar

Figura 9 Ingreso Datos Informativos del Asociado
Fuente: El Autor, 2013

Para ingresar los campos nombramiento, código de perfil, servicio Wap y activo se seleccionan de una lista desplegable (Figura 10).

Figura 10 Listas desplegables para seleccionar información
Fuente: El autor, 2013

Para ingresar el campo cédula se valida el ingreso de este campo ya que es obligatorio (Figura 11).

Figura 11 Campo de Cédula Obligatoria
Fuente: El Autor, 2013

Después de ingresar la información requerida del asociado, haciendo clic en el botón guardar se ha creado y se ha adherido un nuevo miembro de la asociación

Para confirmar, se despliega un mensaje de información que se ha guardado el nuevo usuario (Figura 12).

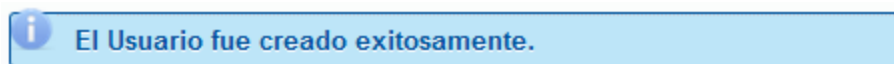


Figura 12 Mensaje de Información de Creación de Usuario
Fuente: El Autor, 2013

⚠ Importante.

Al dar clic en el botón guardar usuario, sin haber llenado los campos, se despliega un mensaje de advertencia que se deben llenar los campos. Estos campos son obligatorios y requeridos, están marcados de color rojo (Fig. 13).

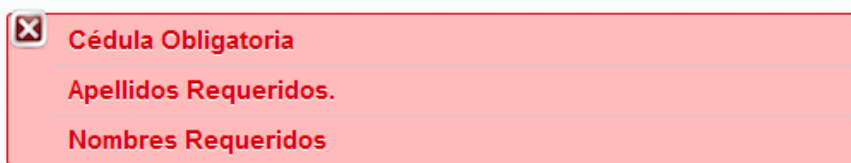


Figura 13 Mensaje Campos Requeridos Y Obligatorios
Fuente: El Autor, 2013

Si requiere buscar el usuario ingresado, haciendo clic en el menú usuario, se despliega la grilla de listado de usuarios, con los campos cédula, nombre, apellido y filtrar como parámetros de búsqueda (Figura 14).

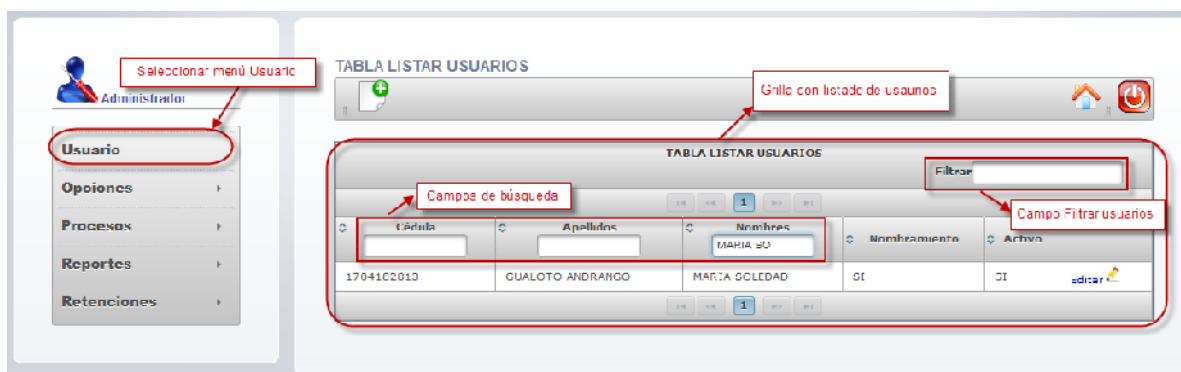


Figura 14 Búsqueda de Usuarios
Fuente: El Autor, 2013

3.2 MENÚ PROCESOS

El menú procesos gestiona el ingreso de módulos, perfiles, así como también ingreso de bancos, registro de cuentas bancarias, préstamos, casa comercial, aportes, cheques y facturas (Figura 15).

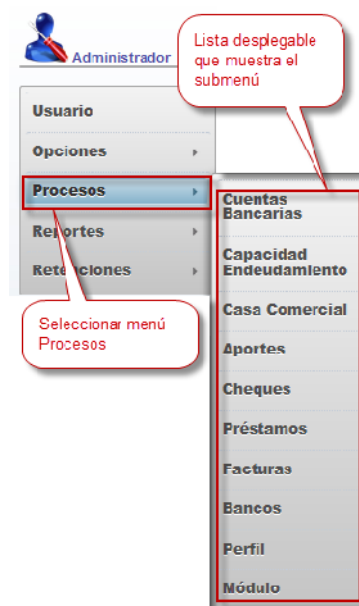


Figura 15 Menú Procesos
Fuente: El Autor, 2013

3.2.1 SUBMENÚ BANCOS

El submenú Bancos muestra una barra de tareas con los botones que tiene como funcionalidad mostrar los bancos con los que se realiza giro de cheques o apertura de cuentas bancarias, agregar registros de nuevos bancos, editar los bancos existentes y eliminarlos, si lo requiere la asociación (Figura 16).

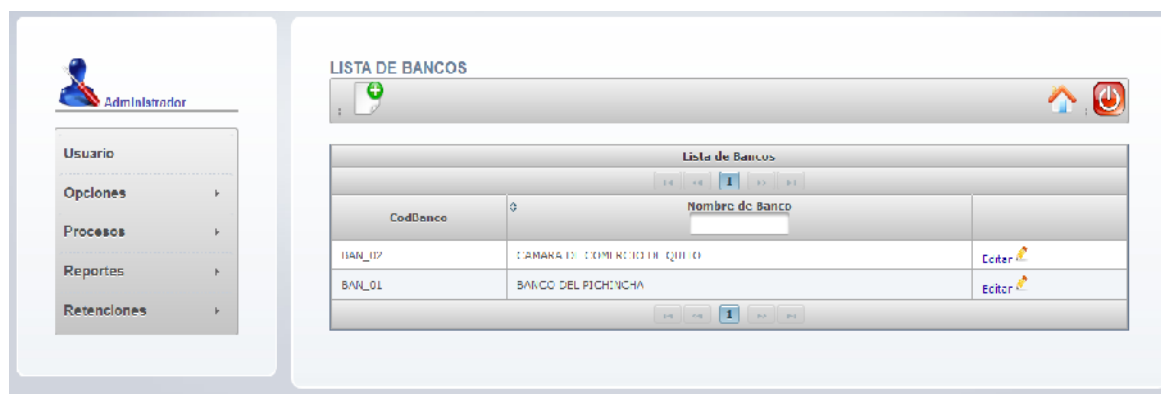


Figura 16 Submenú Bancos
Fuente: El Autor, 2013

En la parte superior de la pantalla se encuentra la barra de tareas con el botón llamado Agregar Banco, tiene como funcionalidad ingresar nuevos bancos que tienen relación de negocio con la asociación, haciendo clic se despliega una grilla para ingresar el nombre del banco, es un campo de texto (Figura 17).

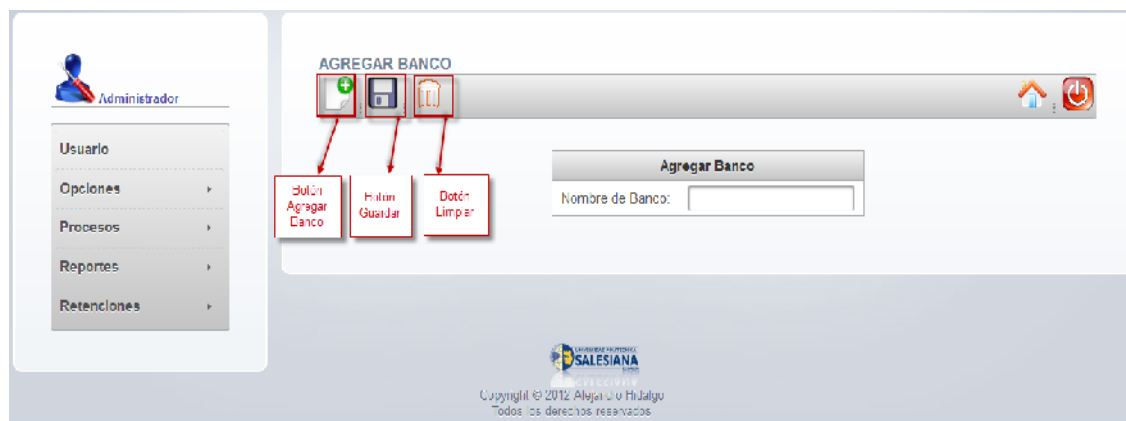


Figura 17 Ventana Agregar Bancos
Fuente: El Autor, 2013

Llenar el campo nombre del banco, seleccionar el botón guardar banco (Figura 18). Se despliega el mensaje de información que se ha agregado un nuevo banco (Figura 19).

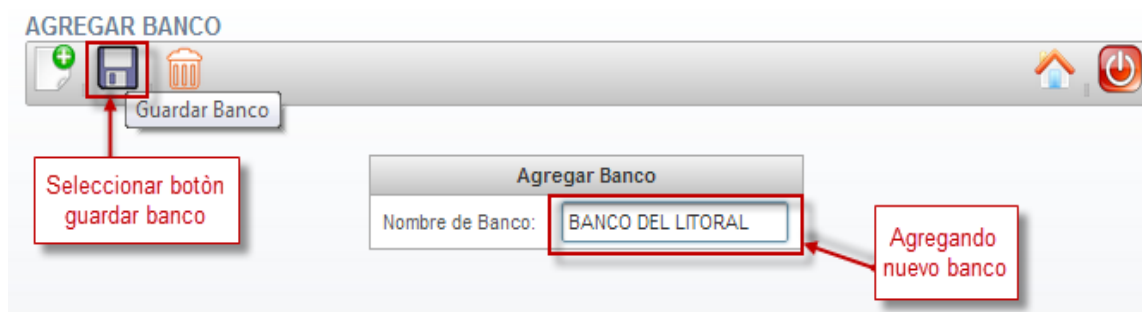


Figura 18 Ingresando Nuevo Banco
Fuente: El Autor, 2013

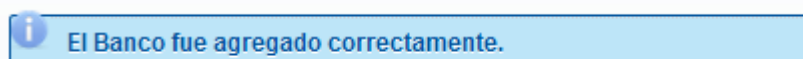


Figura 19 Mensaje de Información
Fuente: El Autor, 2013

3.2.2 SUBMENÚ CUENTAS BANCARIAS

Permite ingresar, listar, modificar y eliminar cuentas bancarias (Figura 20).



Figura 20 Submenú Cuentas Bancarias
Fuente: El Autor, 2013

El botón agregar cuentas, crea nuevas cuentas bancarias, al dar clic, se despliega una grilla con campos de selección como nombre del banco, tipo de cuenta, campos de texto como número de la cuenta corriente

Ingresar los campos de selección y texto, dar clic en el botón guardar cuenta bancaria (Figura 21), para confirmar la acción se despliega un mensaje de información que se ha agregado una nueva cuenta bancaria (Figura 22).

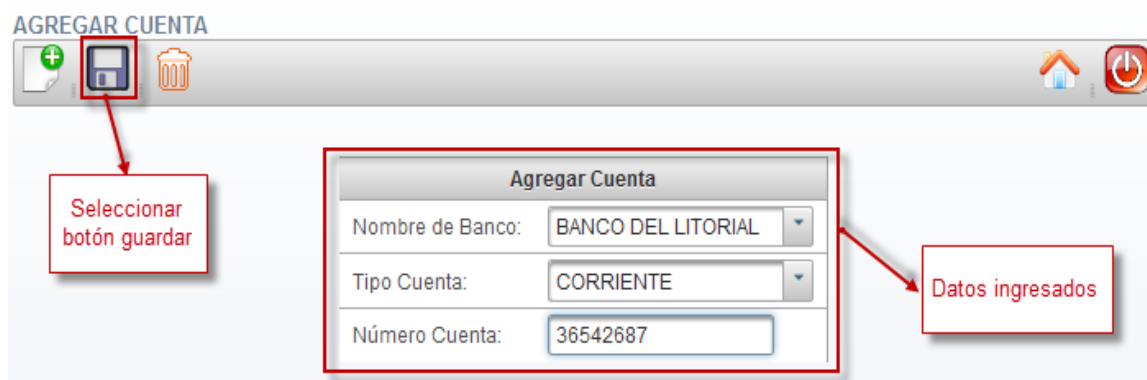


Figura 21 Agregar cuenta bancaria
Fuente: El Autor, 2013

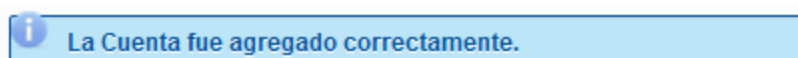


Figura 22 Mensaje de Información Cuenta Agregada
Fuente: El Autor, 2013

3.2.3 SUBMENÚ CAPACIDAD DE ENDEUDAMIENTO

La capacidad de endeudamiento es la opción que nos permite ver si un asociado cumple con los requisitos establecidos por la asociación para que se pueda otorgar un préstamo, ser garante del mismo o saber si puede endeudarse conociendo los pagos mensuales que le serán debitados. Para esto se tiene como antecedentes los siguientes puntos (Figura 23):

- El asociado debe pertenecer más de un mes en la asociación.
- El valor de gastos de la presidencia debe haber sido ingresado.
- El sueldo del asociado debe haber sido ingresado.
- El valor de sueldo y gastos de la presidencia deben ser mayores a 0.

Cálculo de Capacidad de Endeudamiento


Cálculo de Capacidad de Endeudamiento

Usuario

Valor Préstamo

Meses de Préstamo

Interés



//Antecedentes: 1. El asociado debe pertenecer más de un mes en la asociación 2. El valor de gastos de la presidencia debe haber sido ingresado 3. El sueldo del asociado debe haber sido ingresado 4. El valor de sueldo y gastos de la presidencia deben ser mayores a 0

Figura 23 Cálculo Capacidad Endeudamiento
Fuente: El Autor, 2013

Para proceder con un cálculo de capacidad de endeudamiento se llenan los valores dentro de las cajas de texto, con la información necesaria para después dar clic en ícono Calcular Capacidad de Endeudamiento(Figura 24).

Cálculo de Capacidad de Endeudamiento

Usuario: CAJAS DEFAZ EDISON PATRICIO

Valor Préstamo: 300

Meses de Préstamo: 3

Interés: 0.05

Porcentaje 5% - 0.05

Calcular la Capacidad de Endeudamiento

Detallar

Figura 24 Capacidad de Endeudamiento Campos Llenos
Fuente: El Autor, 2013

Después de dar clic en el ícono de Calcular la capacidad de endeudamiento tenemos dos opciones que serán mostradas por el sistema siempre y cuando se cumplan los antecedentes antes mencionados.

3.2.3.1 APTO PARA UN PRÉSTAMO

El Asociado SI ES APTO para un préstamo

Figura 25 Mensaje de Advertencia Apto
Fuente: El Autor, 2013

Si se desea se puede dar clic en el botón Detallar que nos explicará las cuotas a futuro para las cantidades que le serán debitadas (Figura 26).

Detalle Capacidad de Endeudamiento

Sueldo: \$ 920.75 Gasto Presidencia: \$ 250.00

Préstamo: \$ 300.0 Interés: 0.05

Detalle de Préstamo

Meses	Cuota	Saldo
11/marzo/2013	\$ 105.00	\$ 210.00
11/abril/2013	\$ 105.00	\$ 105.00
11/mayo/2013	\$ 105.00	\$ 0.00

Meses que se le retendrá

Imprimir

Cuota a Pagar

Figura 26 Cuotas Préstamo Apto
Fuente: El Autor, 2013

3.2.3.2 NO APTO PARA UN PRÉSTAMO



Figura 27 Mensaje de Advertencia no Apto
Fuente: El Autor, 2013

De igual manera se detalla porque no se puede efectuar un préstamo y nos mostrará el valor por el cual se puede endeudar el asociado (Figura 28).

Detalle Capacidad de Endeudamiento

Sueldo:\$ 920.75	Gasto Presidencia:\$ 250.00
Préstamo:\$ 3000.0	Interés: 0.05

La retención no debe exceder al 70% de su sueldo

Cuota1: \$1050.00 ----- RetenciónTotalAprox: \$1309.00

Puede realizar un préstamo hasta de : \$1101.0
Detallado a continuación para 3 meses

Detalle de Préstamo		
Meses	Cuota	Saldo
11/marzo/2013	\$ 385.35	\$ 770.70
11/abril/2013	\$ 385.35	\$ 385.35
11/mayo/2013	\$ 385.35	\$ 0.00

Próximos meses de Retención

Cuotas a Pagar con el nuevo préstamo

Figura 28 Cuotas Préstamo no Apto
Fuente: El Autor, 2013

3.3 MENÚ OPCIONES

En el menú opciones tenemos la posibilidad de interactuar con otros sistemas como es enviar de SMS, exportar las retenciones generadas mensualmente y también registrar los gastos externos que tiene la asociación.



Figura 29 Menú Opciones
Fuente: El Autor, 2013

3.3.1 ENVIAR SMS

El submenú enviar SMS, envía SMS a los asociados que tengan ingresados correctamente su número de celular que consta actualmente con 10 dígitos, esto se lo puede agregar al crear o modificar un asociado ingresando la información del número de celular. (Vea ítem 3.1). A demás muestra una pantalla donde permite contar con las siguientes opciones:

3.3.1.1 ENVIO SMS DE RETENCIONES

Esta opción indica que las retenciones generadas exitosamente el último mes se enviarán a los destinatarios que se les generaron las retenciones y que cuenten con números de celulares actualizados (Figura 30).

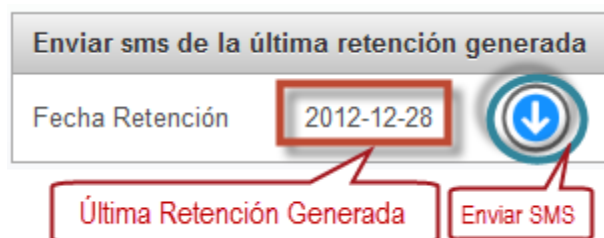


Figura 30 Enviar SMS Retenciones
Fuente: El Autor, 2013

Dar clic en el ícono Enviar SMS. Si no hay retenciones generadas en los últimos 20 días muestra el siguiente mensaje (Figura 31).

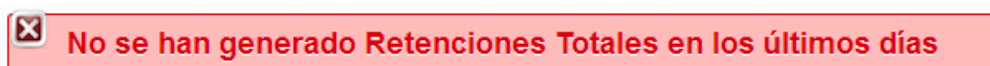


Figura 31 SMS no Retenciones Totales
Fuente: El Autor, 2013

Caso contrario si no hubo ningún error indica el siguiente mensaje (Figura 32).

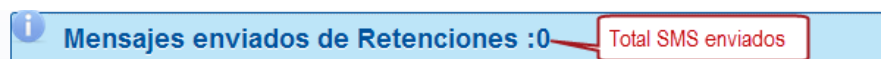


Figura 32 SMS Enviados Retenciones
Fuente: El Autor, 2013

3.3.1.2 ENVIO DE SMS GENERALES

Esta opción permite enviar SMS en general a todos los asociados que tengan su número de celular actualizado, puede usarse para notificar de sesiones u otros acontecimientos que respectan a la asociación (Figura 33).

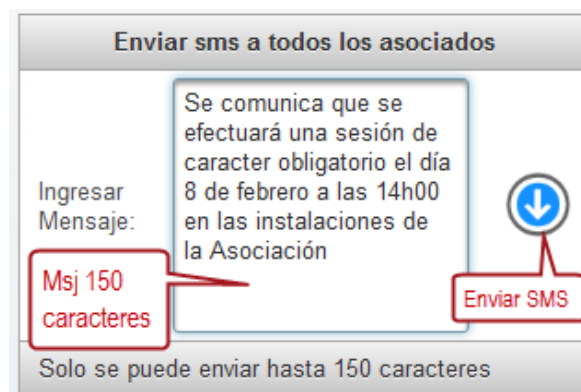


Figura 33 SMS Generales Enviados
Fuente: El Autor, 2013

3.3.2 EXPORTAR ARCHIVO

Exportar un archivo se refiere a las retenciones generadas mensualmente por la asociación que necesitan ser enviadas a la Presidencia de la República del Ecuador para que se haga el trámite respectivo en el rol de pagos y se ha decidido contar con varios archivos de exportación más usados en la actualidad (Figura 34).

GENERACIÓN DE ARCHIVO DE RETENCIONES

▼ Exportar A:

Generación de archivo de Retenciones

FechaActual: 11/febrero/2013

Seleccione la Fecha: 🔍

Seleccione la fecha de las retenciones para generar el archivo

TABLA ARCHIVO MENSUAL DE RETENCIONES TOTALES

Cédula	Apellido	Nombre	MontoTotal
No records found.			

Figura 34 Exportar Archivo

Fuente: El Autor, 2013

Se debe seleccionar una fecha del menú desplegable “*Seleccione una Fecha*”, que corresponden a las retenciones ya generadas, para luego dar clic en el ícono buscar (lupa), y mostrará una lista con la información necesaria para ser exportada (Figura 35).

Generación de archivo de Retenciones

FechaActual: 11/febrero/2013

Seleccione la Fecha: 🔍 **Buscar**

Consulta devuelta para exportar

TABLA ARCHIVO MENSUAL DE RETENCIONES TOTALES

Cédula	Apellido	Nombre	MontoTotal
1704079480	ZULETA ENRIQUEZ	CARLOS GUSTAVO	9.00
0501482137	VIZUETE TOAPANTA	DAMARIZ ASUNCION	9.00
1232323433	SOÑÍA LEPÚ	TÍE NUÑEZ Í Á	9.00
0400429544	REVELO SIERRA	GJILLERMO WUGNER	9.00
0600806343	QUINTANILLA ALMENDARIZ	JULIO ENRIQUE	111.00
3333333333	PRUEBA3		16.00
1717171717	PRIIFRA1	PRIIFRA11	9.00
1103191639	ORDOÑEZ	FLOR	16.00
0501190310	MOLINA I IERRERA	R:GODERTO I IIPOLITO	9.00
0601933427	LOPEZ RUBIO	JAIME EDISON	19.00
1713066341	LEON	CHRISTIAN	9.00
1704182813	GUALOTO ANDRANGO	MARIA SOLEDAD	9.00

Figura 35 Consulta Devuelta

Fuente: El Autor, 2013

Con esta información se procede a exportar la información que corresponde a la retención total generada por el sistema a los diferentes formatos de archivos, con la ayuda del menú Exportar (Figura 36).



Figura 36 Exportar Información
Fuente: El Autor, 2013

3.3.3 EGRESOS

La opción de Egresos permite registrar gastos propios de la asociación como son: pagos de servicios básicos, pagos a terceros o registro egresos en general que se llevan (Figura 37).

TABLA CUENTAS GENERALES ASOCIACIÓN

TABLA CUENTAS GENERALES ASOCIACIÓN					
CodCta	CasaComercial	Tipo	Monto	Fecha	
CTA_0003	CTAS GENERALES ASOCIACION	PAGO LUZ	23.45	02/agosto/2012	Editar
CTA_0002	CTAS GENERALES ASOCIACION			02/agosto/2012	Editar
CTA_0001	CTAS GENERALES ASOCIACION	PAGO LUZ	23.05	27/julio/2012	Editar

Figura 37 Lista de Egresos
Fuente: El Autor, 2013

3.4 MENÚ REPORTES

El menú reportes se encarga de buscar las retenciones para generar los reportes (Figura 38)



Figura 38 Menú Reportes
Fuente: El Autor, 2013

3.4.1 SUBMENÚ MENSUAL

El submenú mensual permite seleccionar las fechas de las retenciones para generar un reporte mensual (Figura 39).

Para ello:

- Seleccionar la fecha de las retenciones.
- Escoger el botón buscar.

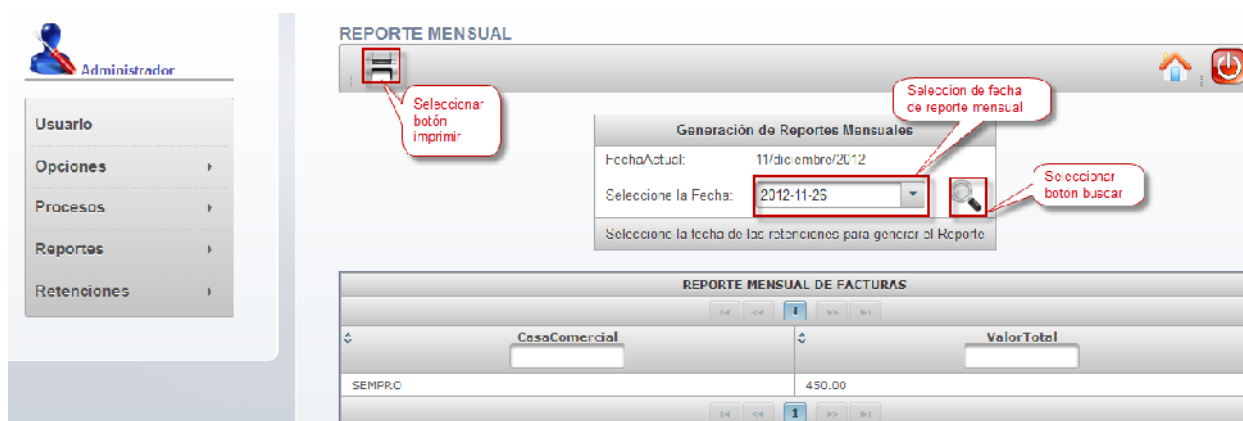


Figura 39 Buscar Reporte Mensual
Fuente: El Autor, 2013

Seguidamente, se muestra en tres listas llamadas:

- Reporte mensual de facturas.

- Reporte mensual de aportes.
- Reporte mensual de préstamos.

Estas listas permite conocer cuáles son las retenciones totales de cada una de los reportes (Figura 40), con los valores totales detallados por:

- Aportes generados.
- Casas comerciales.
- Prestamos debitados.

REPORTE MENSUAL DE FACTURAS	
<div> <div>CasaComercial</div> <div></div> </div>	<div> <div>ValorTotal</div> <div></div> </div>
SEMPRO	450.00
REPORTE MENSUAL DE APORTES	
<div> <div>Aporte</div> <div></div> </div>	<div> <div>ValorTotal</div> <div></div> </div>
SISTEMA CELULAR	3.00
SISTEMA CELULAR NO ASOCIADO	10.00
AYUDA MAYRA VILLABAS	0.00
PAGO WEB	3.50
ASOCIACION ASOCIADO	5.00
ASOCIACION NO ASOCIADO	20.00
REPORTE MENSUAL DE PRESTAMOS	
<div> <div>Préstamo</div> <div></div> </div>	<div> <div>ValorTotal</div> <div></div> </div>
Prestamos Mensuales:	102.77

Figura 40 Lista de Reportes de Retenciones Totales de Aportes Generados, Casas Comerciales y Préstamos debitados

Fuente: El Autor, 2013

Acciones:

Botón Imprimir: permitirá imprimir el reporte mostrado.

Para ellos, previamente se debe realizar la búsqueda de los reportes totales, luego dar clic en el botón imprimir (Figura 41).

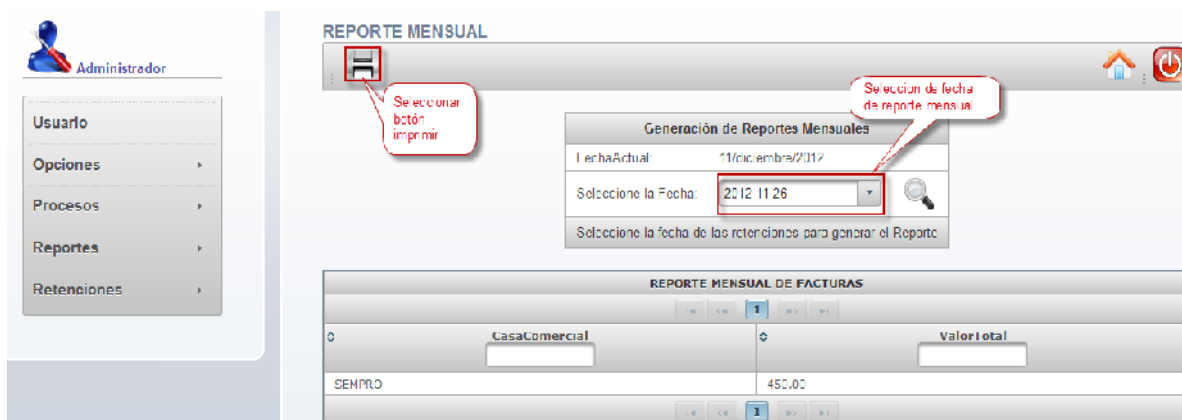


Figura 41 Botón Imprimir
Fuente: El Autor, 2013

3.5 MENÚ RETENCIONES

El menú de retenciones es la parte principal del sistema al ser de gran importancia por los procesos que se generan automáticamente se debe tener mucho cuidado al manipular esta clase de información, que representa a: préstamos, aportes y facturas, que es lo que se da en la asociación para al final tener un proceso que reúne los anteriores para generar una retención total de acuerdo al asociado y a la fecha de corte (Figura 42).

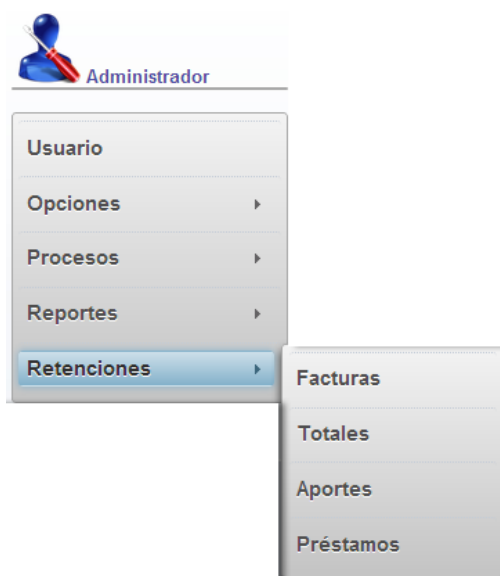







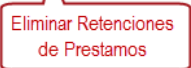
Figura 42 Menú Retenciones
Fuente: El Autor, 2013


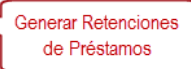

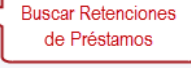
3.5.1 SUBMENÚ PRÉSTAMOS

El submenú préstamos nos muestra la opción de generar todos los préstamos pendientes para el mes en curso, buscar retenciones de préstamos generadas de meses anteriores, agregar una nueva retención de forma individual, modificar una retención si la fecha de creación no sobrepasa con 15 días a la fecha actual, ya que en ese tiempo las retenciones pasan a la Presidencia y por último nos permite eliminar un grupo de préstamos generados en una fecha determinada (Figura 43).

LISTA RETENCIONES DE PRÉSTAMOS

 Nueva Retención de Préstamo
  Eliminar Retenciones de Prestamos

Generar las Retenciones de Préstamo para el mes en curso
 FechaActual: 14/febrero/2013   Generar Retenciones de Préstamos
 Buscar Retenciones de Prestamos: Seleccione una Fecha   Buscar Retenciones de Préstamos
 Las retenciones se generan solo una vez al mes

LISTA RETENCIONES DE PRÉSTAMOS

CodRetPres	IdUsuario	CodPrestamo	MontoRetPres	FechaRetPres
No records found.				

Figura 43 Retenciones de Préstamos
Fuente: El Autor, 2013

3.5.1.1 GENERAR RETENCIONES DE PRÉSTAMOS

Para generar las retenciones de préstamos automáticamente se presiona el ícono *Generar Retenciones de Préstamos*, y si todo está correcto nos muestra los siguientes mensajes (Figura 44). Estas retenciones corresponden a préstamos que se deben cobrar para el mes en curso sean préstamos que se han generado en meses anteriores que aún siguen vigentes y nuevos préstamos que todavía no se han cobrado.

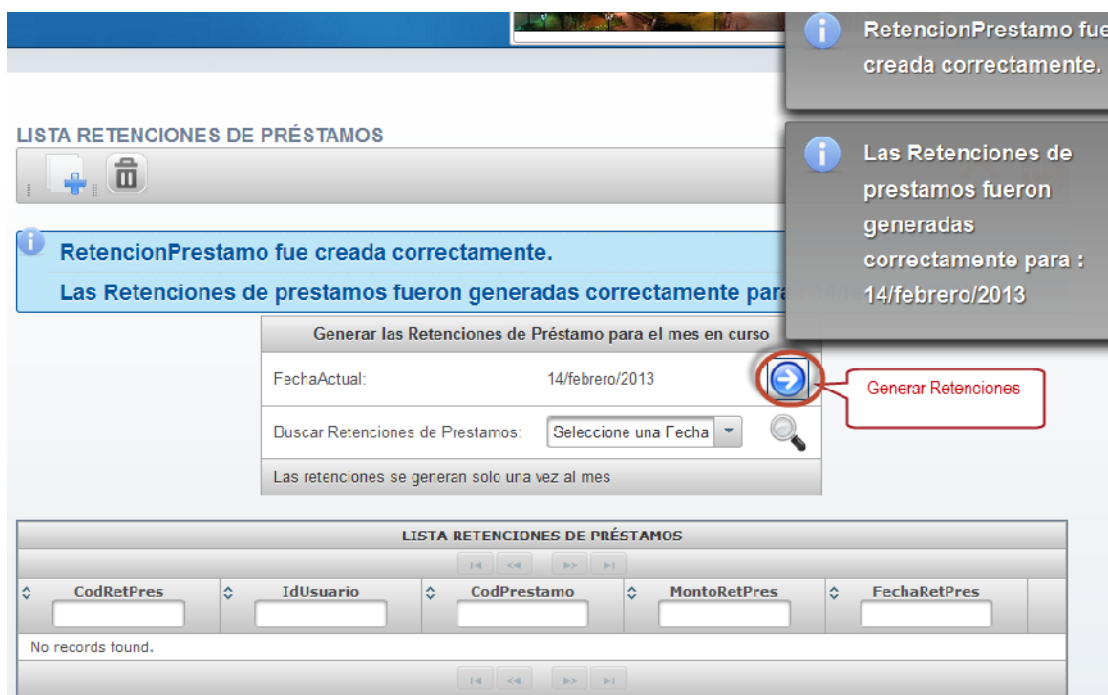


Figura 44 Retenciones de Préstamos Generar
Fuente: El Autor, 2013

3.5.1.2 BUSCAR RETENCIONES DE PRÉSTAMOS

Esta opción sirve para buscar retenciones de préstamos que ya han sido generadas, ya sea por alguna consulta o duda que se tenga se debe tener en cuenta que si se desea editar alguna retención manualmente la fecha en que se generó no deberá ser menor a 15 días de la fecha actual.

Para buscar se selecciona la fecha de un menú desplegable que muestra las fechas de las últimas retenciones y se da clic en el ícono buscar (Figura 45).

LISTA RETENCIONES DE PRÉSTAMOS

Generar las Retenciones de Préstamo para el mes en curso

FechaActual: 14/febrero/2013

Buscar Retenciones de Prestamos: 2013-02-14

Las retenciones se generan solo una vez al mes

LISTA RETENCIONES DE PRÉSTAMOS

CodRetPres	IdUsuario	CodPrestamo	MontoRetPres	FechaRetPres	
RPR_0005	1212121219	PRE_0002	103.00	14/febrero/2013	Editar

Figura 45 Retenciones de Préstamos Buscar
Fuente: El Autor, 2013

3.5.1.3 ELIMINAR RETENCIONES DE PRÉSTAMOS

Para eliminar las retenciones de préstamos generadas anteriormente, solo se muestra la última retención generada, manteniendo una norma de seguridad, que indica que la información anterior a la última no se la pueda realizar cambios. Se procede a dar clic en el ícono *Eliminar Retenciones de Préstamos*, seleccionamos la fecha cuando se realizó la última retención editable y presionar el ícono de basurero *Eliminar Retenciones de Préstamos*, si todo está correcto nos muestra el mensaje indicado (Figura 46).

ELIMINACIÓN DE LAS RETENCIONES DE PRESTAMOS

Las Retenciones de prestamos han sido eliminadas para : 2013-02-14

Las Retenciones de prestamos han sido eliminadas para : 2013-02-14

Eliminación de las Retenciones de Prestamos

FechaActual: 14/febrero/2013

Seleccione la Fecha: Seleccione una Fecha

Seleccione la fecha de los prestamos que se eliminarán.

Eliminar Retenciones de Préstamos

Figura 46 Retenciones de Préstamos Eliminar
Fuente: El Autor, 2013

3.5.2 SUBMENÚ FACTURAS

El submenú facturas permite crear nueva retenciones de facturas, las cuales generan el descuento de una casa comercial para un asociado, se mostrará un menú con una barra de tareas con los botones agregar nueva retención y buscar retenciones con la factura de la casa comercial en una lista desplegable, los asociados ordenados por apellido, el monto y fecha de la retención.

Acciones

Botón Agregar nueva retención: permite registrar nuevas retenciones de facturas.

Para ello, dando clic en el botón agregar nueva retención, se despliega una lista con campos de llenado para ingresar los datos de la retención como:

- Código de factura: permite seleccionar un código de factura de las diferentes casas comerciales (Figura 47).
- Nombre de usuario: permite seleccionar el asociado, de quien se va ingresar el valor de los gastos (Figura 48).
- Monto de la retención: permite ingresar el monto de retención de la factura.
- Fecha de la retención: permite seleccionar la fecha de la retención (Figura 49).

Después de ingresar y seleccionar los datos, dar clic en el botón guardar retención de factura (Figura 50).

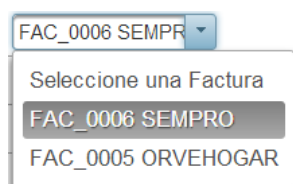


Figura 47 Selección código de factura
Fuente: El Autor, 2013

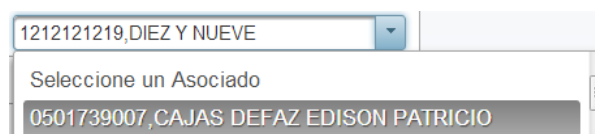


Figura 48 Selección del asociado
Fuente: El Autor, 2013

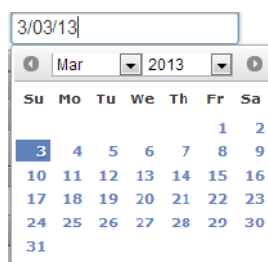


Figura 49 Selección de la fecha de la retención
Fuente: El Autor, 2013

 A screenshot of a web application interface titled 'AGREGAR RETENCIÓN DE FACTURA'. The form contains several input fields: 'CodRetencion_Factura:', 'Cod_Factura:' (with a dropdown menu showing 'FAC_0006 SEMPR'), 'IdUsuario:' (with a dropdown menu showing 'CAJAS DEFAZ EDISON PATRICIO'), 'MontoRetencion:' (with the value '500'), and 'FechaRetencion:' (with the date '3/03/13'). A red box highlights the form fields, and a callout bubble points to the 'Guardar' (Save) button with the text 'Seleccionar botón guardar retención'. Another callout bubble points to the 'Cod_Factura' dropdown with the text 'Ingreso de datos de la retención de la factura'. Below the form is a table titled 'LISTA RETENCIONES DE FACTURA' with columns: 'RetencionFact', 'IdUsuario', 'Cod_Factura', 'MontoRetencion', and 'FechaRetencion'. The table is currently empty, showing 'No records found.'

Figura 50 Ingreso de datos de Retención
Fuente: El Autor, 2013

Posteriormente se despliega un mensaje de advertencia que se ha ingresado satisfactoriamente el registro de retención de factura (Figura 51).

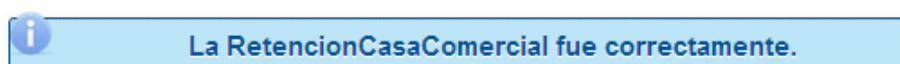


Figura 51 Mensaje de advertencia de retención
Fuente: El Autor, 2013

Botón Buscar retención factura: permite realizar una búsqueda de retenciones de facturas de acuerdo a una fecha de inicio y fin.

Para ello, dando clic en el botón buscar retención de factura, se despliega un ventana con una lista para seleccionar las fecha, un botón para generar las retenciones, al presionar el botón se ejecuta la búsqueda (Figura 52).

BUSCAR RETENCIONES DE FACTURAS

Nueva Retención de Factura

Buscar Retenciones de Facturas

FechaDesde: 09/10/2012

FechaHasta: 03/03/2013

Las retenciones se generan solo una vez al mes

Selección de las fechas de búsqueda

Seleccíonar botón generar retención

LISTA RETENCIONES DE FACTURA

CodRetencion_Factura	IdUsuario	Cod_Factura	MontoRetencion	FechaRetencion
No records found.				

Figura 52 Selección de Fechas de Generación de Retención
Fuente: El Autor, 2013

Posteriormente se despliega la búsqueda en una lista, al final de la lista se encuentra un botón para editar el registro de la retención (Figura 53). Los parámetros son:

- Código de retención de factura
- Código del asociado
- Código de la factura
- Monto de retención
- Fecha de retención

BUSCAR RETENCIONES DE FACTURAS

FechaDesde: 09/10/2012

FechaHasta: 03/03/2013

Las retenciones se generan solo una vez al mes

LISTA RETENCIONES DE FACTURA

Filtros de búsqueda

Botón Editar

CodRetencion_Factura	IdUsuario	Cod_Factura	MontoRetencion	FechaRetencion	Editar
RFC_0010	0501739007	FAC_0006	500.00	03/marzo/2013	Editar
RFC_0001	0501739007	FAC_0001	10.00	17/octubre/2012	Editar
RFC_0008	1212121219	FAC_0005	10.00	24/diciembre/2012	Editar
RFC_0007	1212121219	FAC_0003	60.00	20/noviembre/2012	Editar
RFC_0002	1212121219	FAC_0001	20.00	22/octubre/2012	Editar
RFC_0011	0601614324			02/marzo/2013	Editar
RFC_0006	0601614324	FAC_0004	20.00	20/noviembre/2012	Editar
RFC_0005	0601614324	FAC_0004	50.00	20/noviembre/2012	Editar
RFC_0009	0601933427	FAC_0005	10.00	26/diciembre/2012	Editar
RFC_0003	1704182813	FAC_0002	10.00	15/octubre/2012	Editar
RFC_0004	0501190318	FAC_0002	10.00	15/octubre/2012	Editar

Registros de retención de factura

Figura 53 Búsqueda ejecutada de las retenciones

Fuente: El Autor, 2013

3.5.3 SUBMENÚ APORTES

El submenú aportes permite crear nueva retenciones de aportes, las cuales generan los aportes propios mensuales de la asociación que deben ser cancelados, así como los aportes por gastos varios. Todo esto se generará por cada uno de los socios. Además se mostrará un menú con una barra de tareas con los botones nueva retención de aportes y eliminar retención de aportes. Se debe seleccionar la fecha de las retenciones (Figura 54).

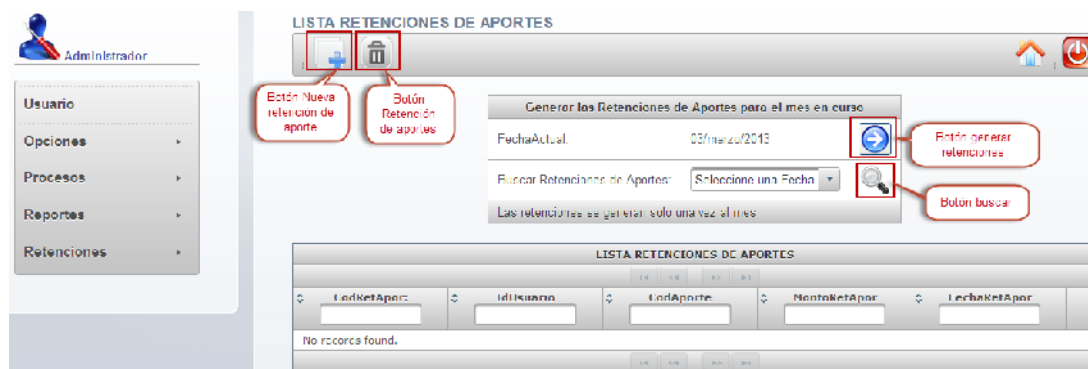


Figura 54 Submenú Aportes
Fuente: El Autor, 2013

Acciones:

Botón Nueva retención de aportes: permite generar retención de aportes mensuales por cada asociado.

Para ello, se debe seleccionar la fecha de la retención, dando clic en la lista desplegable de fechas, la cual debe ser escogida quince días después de haberse generado. Luego se da clic en el botón generar retenciones (Figura 55).

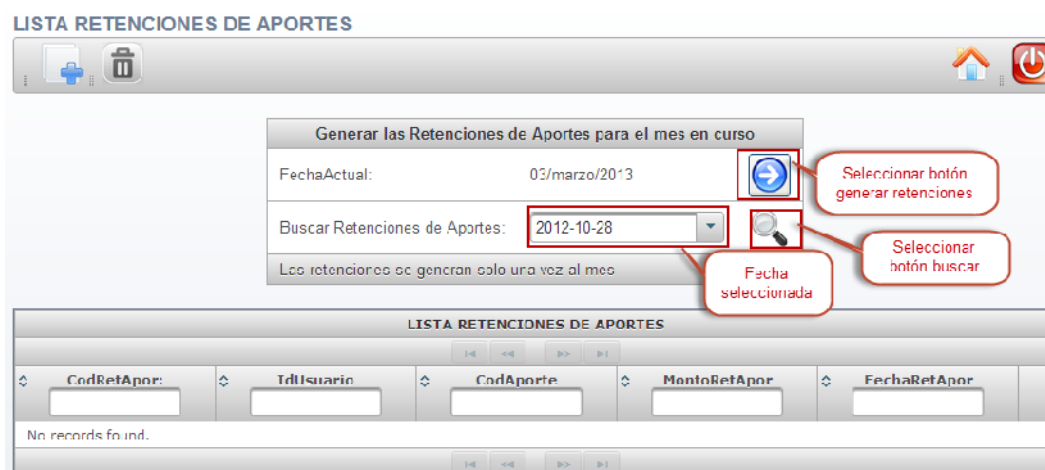


Figura 55 Petición de Generación retención de aportes
Fuente: El Autor, 2013

Posteriormente se despliega un mensaje de advertencia que indica que se ha realizado exitosamente la generación de retención de aportes (Figura 56).

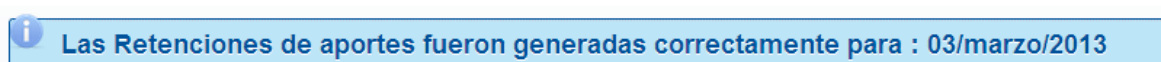
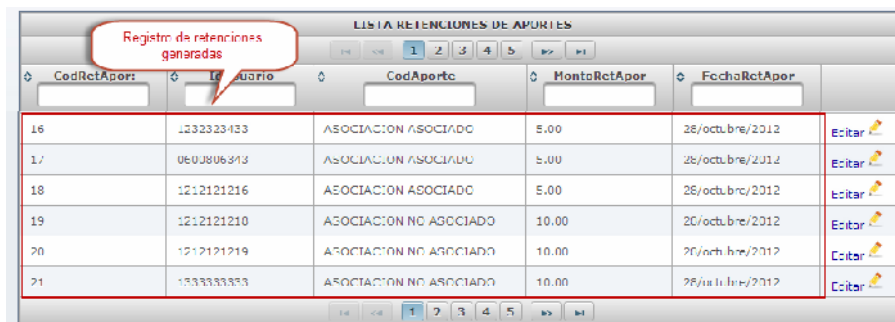


Figura 56 Mensaje de Advertencia
Fuente: El Autor, 2013

Luego dando clic en el botón buscar se ejecuta la generación y se despliega las retenciones de aportes generadas por usuarios (Figura 57).



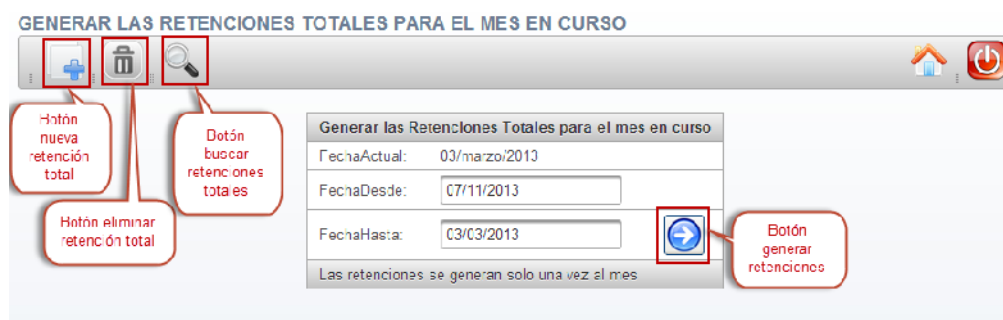
Registro de retenciones generadas

	CodRetApor	CodAporte	MontoRetApor	FechaRetApor	
16	1232333333	ASOCIACION ASOCIADO	5.00	26/octubre/2012	Editar
17	0600806313	ASOCIACION ASOCIADO	5.00	26/octubre/2012	Editar
18	1212121216	ASOCIACION ASOCIADO	5.00	26/octubre/2012	Editar
19	1212121210	ASOCIACION NO ASOCIADO	10.00	26/octubre/2012	Editar
20	1212121219	ASOCIACION NO ASOCIADO	10.00	26/octubre/2012	Editar
21	1233333333	ASOCIACION NO ASOCIADO	10.00	26/octubre/2012	Editar

Figura 57 Generación retención de aportes
Fuente: El Autor, 2013

3.5.4 SUBMENÚ TOTALES

El submenú totales permitirá genera el valor de descuento por cada asociado, se deberá seleccionar las fechas que corresponden al corte del descuento. Además puede buscar las retenciones totales, mostrar una lista desplegable de las fechas que se han realizado las retenciones. Para generar la retención seleccionar la fecha de inicio y de finalización de retención, luego dar clic en el botón generar retenciones (Figura 63).



GENERAR LAS RETENCIONES TOTALES PARA EL MES EN CURSO

Botón nueva retención total

Botón eliminar retención total

Botón buscar retenciones totales

Botón generar retenciones

Generar las Retenciones Totales para el mes en curso

FechaActual: 03/marzo/2013

FechaDesde: 07/11/2013

FechaHasta: 03/03/2013

Las retenciones se generan solo una vez al mes

Figura 58 Submenú Totales
Fuente: El Autor, 2013

Posteriormente se despliega un mensaje de advertencia que se ha generado exitosamente la generación de retenciones totales para el mes en curso (Figura 59).

Las Retenciones Totales fueron generadas correctamente para : 2013-03-04

Figura 59 Mensaje de advertencia
Fuente: El Autor, 2013

Acciones:

Botón nueva retención total: permite registrar manualmente una nueva retención total.

Para ello, seleccionar el nombre del asociado, la fecha de la retención, la fecha de inicio y de finalización, ingresar el valor total para la generación de las retenciones, luego dar clic en el botón guardar (Figura 60).

AGREGAR RETENCIÓN TOTAL

Seleccionar botón guardar

Datos ingresados para registrar nueva retención

Agregar Retención Total	
CodRetencion:	0
IdUsuario:	0501482137, VIZUETE TOAPANTA DAMA
ValorTotal:	20
FechaRetencion:	03/03/2013
FechaDesde:	01/03/2013
FechaHasta:	20/03/2013

Figura 60 Agregar Retención Total
Fuente: El Autor, 2013

Posteriormente, se despliega un mensaje de advertencia que se ha registrado exitosamente la retención total (Figura 61).



Figura 61 Mensaje de advertencia
Fuente: El Autor, 2013

Botón buscar retenciones totales: permite buscar las retenciones totales generadas de cada uno de los meses en curso.

Para ello, se selecciona las fechas de inicio y finalización para buscar las retenciones y generarlas, luego dar clic en el botón buscar (Figura 62).

BUSCAR RETENCIONES TOTALES GENERADAS POR MES

Selección de fechas

Seleccionar botón buscar

Buscar Retenciones Totales generadas por mes	
FechaActual:	03/marzo/2013
Buscar Retenciones Totales:	2013-03-03

Las retenciones se generan solo una vez al mes

Figura 62 Buscar Retenciones de Facturas
Fuente: El Autor, 2013

Seguidamente, se despliega en una lista la generación de retenciones totales (Figura 63).

CodReten	IdUsuario	ValorTotal	03/marzo/2013	01/marzo/2013	10/marzo/2013	
130	1717171717	9.00	03/marzo/2013	01/marzo/2013	10/marzo/2013	Editar
138	3333333333	16.00	03/marzo/2013	01/marzo/2013	10/marzo/2013	Editar
128	1704182813	9.00	03/marzo/2013	01/marzo/2013	10/marzo/2013	Editar
117	1718066341	9.00	03/marzo/2013	01/marzo/2013	10/marzo/2013	Editar
140	0501190318	9.00	03/marzo/2013	01/marzo/2013	10/marzo/2013	Editar
139	0201003175	9.00	03/marzo/2013	01/marzo/2013	10/marzo/2013	Editar

Figura 63 Registros de retenciones totales generados

Fuente: El Autor, 2013

4 MÓDULO SECRETARIA

El módulo secretaria permite administrar el registro de datos e información que se genera cada mes por cada asociado. Entre estas actividades se encuentran:

- Registros de usuarios
- Ingreso de préstamos; casa comercial; aportes; cheques y facturas.
- Registro de retenciones de facturas.
- Registro de egresos de la asociación
- Exportación de archivos

4.1 MENÚ USUARIOS

Permite registrar el ingreso de nuevos usuarios, así como también el restablecimiento de la contraseña.

Para ello:

- Dar clic en el menú usuarios
- Seleccionar botón nuevo usuario

Posteriormente, se despliega una ventana con una lista para ingresar y seleccionar los datos del nuevo socio, luego dar clic en el botón guardar (Figura64).

NUEVO USUARIO

Seleccionar botón guardar

Nuevo Usuario

Cédula	1578253666	Apellidos:	MORETA MENA
Nombres:	WILMER SANTIAGO	Password:	
Dirección:	ELOY ALFARO N45	FonoDomicilio:	2345115
Celular:	056265090	ServicioWap:	SL
Nombramiento:	SI	Activo:	SI
Sueldo:	800	Cargo:	ERVIDOR PUBLICO 1
CodPerfil:	ASOCIADO		

Datos ingresados del nuevo asociado

Figura 64 Datos del nuevo asociado

Fuente: El Autor, 2013

Seguidamente se despliega un mensaje de información que se ha ingresado exitosamente un nuevo socio (Figura65).



Figura 65 Mensaje de Información

Fuente: El Autor, 2013

4.2 MENÚ OPCIONES

El menú opciones permite ingresar y registrar los egresos que tenga la asociación cada mes. Además todos estos registros tienen la opción de ser exportados en archivos, dependiendo de la necesidad del archivo se puede escoger el tipo de extensión.

4.2.1 SUBMENÚ EGRESOS

El submenú egreso permite registrar los egresos que se generen mensualmente.

Para ello:

- Dar clic en el menú opciones
- Seleccionar submenú egresos.

Posteriormente se despliega una ventana con una lista que muestra las cuentas generales de la asociación (Figura66).

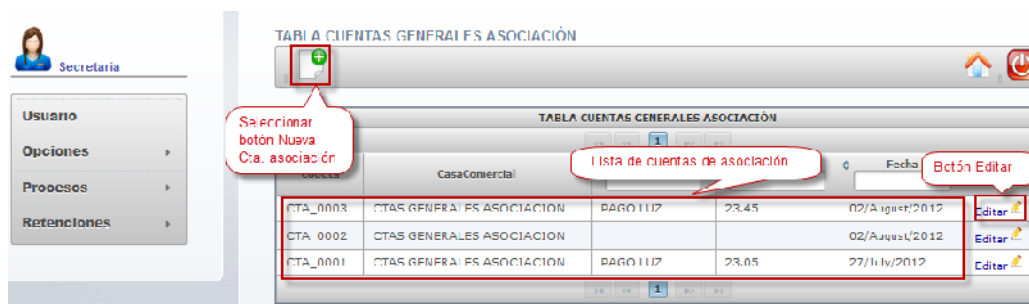


Figura 66 Submenú Egresos
Fuente: El Autor, 2013

Acciones

Botón Nueva cuenta asociación: permite ingresar nuevas cuenta que genera mes a mes la asociación. Para ello, dar clic en el botón nuevo cuenta asociación, se despliega una ventana con una lista para ingresar los datos de una cuenta de asociación. Para este caso los parámetros son:

- Tipo de cuenta: es una cuenta general de la asociación.
- Tipo de la cuenta: descripción de la cuenta
- Monto de la cuenta: valor monetario de la cuenta.
- Fecha de la cuenta: fecha en que se realizó la cuenta

Seguidamente dar clic en el botón guardar (Figura67).

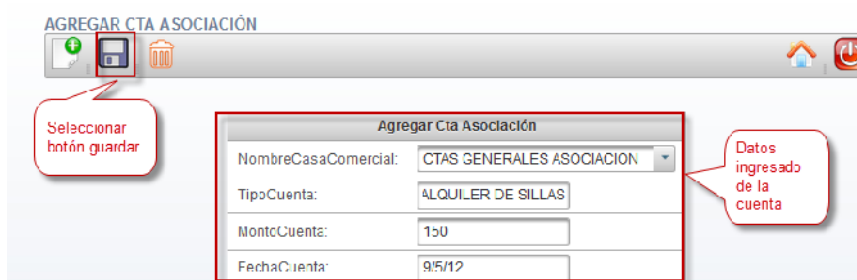


Figura 67 Agregar cuenta de la asociación
Fuente: El Autor, 2013

Luego se despliega un mensaje de advertencia que se ha guardado exitosamente una cuenta general de la asociación (Figura68).

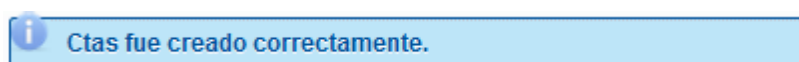


Figura 68 Mensaje de advertencia
Fuente: El Autor, 2013

4.2.2 SUBMENÚ EXPORTAR

El submenú exportar permite enviar la generación de archivos de retenciones en un archivo con el tipo de extensión .pdf, .cvs y .xml (Figura 69).

Para ello:

- Se define el tipo de extensión del archivo.
- Seleccionar la fecha de la generación de las retenciones.
- Escoger el botón buscar.

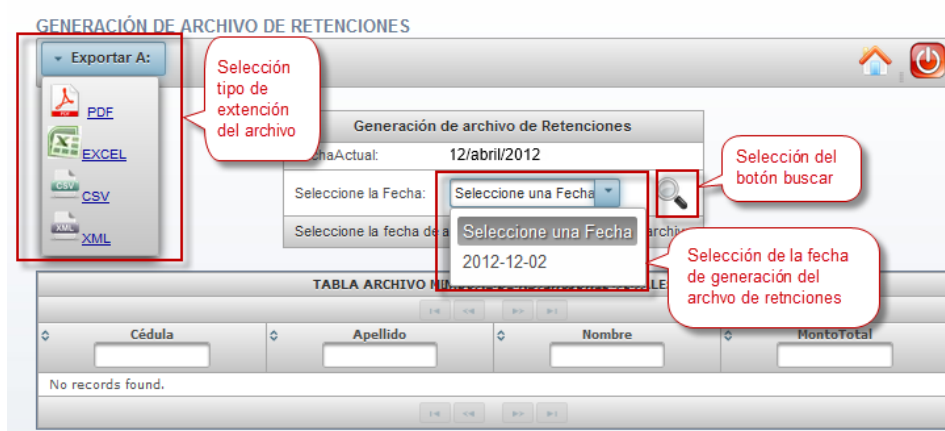


Figura 69 Generación del Archivo de Retenciones
Fuente: El Autor, 2013

Posteriormente se muestra en la lista los registros generados de las retenciones (Figura 70).

Cédula	Apellido	Nombre	MontoTotal
133333333	LOPEZ RIVAS	MARIA ANGELICA	60.00
0400429544	REVELO SIERRA	GUILLERMO WUGNER	123.50
1212121219	DIEZ Y NUEVE		15.77

Figura 70 Búsqueda de Registros Generados
Fuente: El Autor, 2013

4.3 MENÚ PROCESO

4.3.1 SUBMENÚ PRÉSTAMOS

Registra los préstamos que se les otorga a los socios, modifica y elimina préstamos ingresados.

Seleccionar botón agregar préstamo, se despliega un ventana con una grilla de campos de texto y de selección para ser llenados como asociado, número de cheque, fecha de préstamos y número de cuotas (Figura 71). Dar clic en el botón guardar, muestra un mensaje de información de registro agregado (Figura 72).

The screenshot shows a web form for adding a loan. It includes several input fields and a list of associated persons. Red boxes and arrows highlight specific elements:

- Selección de Cheque:** A dropdown menu labeled 'Seleccione un Cheque' with the value '002501' selected. An annotation 'Seleccionar número de cheque' points to it.
- Selección de Asociado:** A dropdown menu labeled 'Seleccione un Asociado' with a list of names and IDs. The entry '0400429544, REVELO SIERRA GUILLERMO WUGNER' is highlighted. An annotation 'Seleccionar asociado' points to it.
- Número de Cuotas:** A text input field labeled 'Cuotas:' with the value '2'. An annotation 'Ingresar número de cuotas' points to it.
- Fecha de Préstamo:** A date picker labeled 'FechaPréstamo:' showing the date '7/12/12'. An annotation 'Ingresar fecha de préstamo' points to it.

Figura 71 Campos de Agregar Préstamo
Fuente: El Autor, 2013

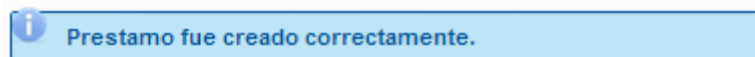


Figura 72 Mensaje de Información Préstamo Agregado
Fuente: El Autor, 2013

4.3.2 SUBMENÚ CASA COMERCIAL

Permite ingresar las casas comerciales con las que se ha acordado descuentos de compra para los socios, así como modificar o eliminar las casas.

Seleccionar botón agregar casa comercial, se despliega una grilla con campos de texto y selección, llenar estos campos y dando clic en el botón guardar (Figura 73) se despliega mensaje de advertencia que se agregó la casa comercial (Figura 74).

NUEVA CASA COMERCIAL

Seleccionar botón guardar

Campos de casas comercial llenados

Nueva Casa Comercial	
NombreCc:	CORPORACIÓN LA FAVO
NombreContacto:	FRANCISCO LOPEZ
TelefonoContacto:	09875612464
FechaInicioRelacion:	1/11/12
Carnet:	SI
Descuento:	10%

Figura 73 Campos de Casa Comercial
Fuente: El Autor, 2013

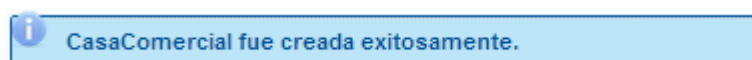


Figura 74 Mensaje de Advertencia Casa Comercial
Fuente: El Autor, 2013

4.3.3 SUBMENÚ APORTES

El submenú Aportes permite registrar los aportes mensuales que tienen los socios. Además de modificar, actualizar o eliminar. Haciendo clic en botón agregar aportes se despliega una lista (Figura 75) con campos de llenado y selección para ser llenados como:

- Nombre de aporte: campo requerido.
- Monto: valor del aporte, campo requerido.
- Nombramiento: si posee o no nombramiento.
- Activo: si es un socio activo o no.
- Detalle del aporte: descripción del aporte.
- Meses del aporte: valor numérico de meses de aporte.

Figura 75 Agregar Aporte Mensual

Fuente: El Autor, 2013

Luego, haciendo clic en el botón guardar se despliega un mensaje de información de ingreso satisfactorio del cheque (Figura 76).

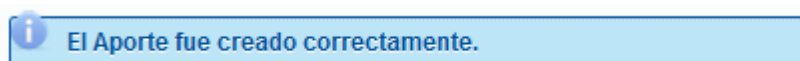


Figura 76 Mensaje de Información Aporte

Fuente: El Autor, 2013

4.3.4 SUBMENÚ FACTURAS

Permite ingresar las facturas generadas por las casas comerciales, acerca de los valores mensuales de los socios como son los gastos y descuentos (Figura77).

Figura 77 Submenú Facturas

Fuente: El Autor, 2013

Seleccionar el botón agregar factura se despliega una con campos de selección como nombre de la casa comercial, número de cheque, campos de textos fecha

de factura, subtotal, monto total, estos campos son obligatorios, total a pagar, RUC, número de factura, autorización, fecha máxima de pago y gana asociado (Figura 78).

Figura 78 Campos de Factura

Fuente: El Autor, 2013

Seleccionando el botón guardar, se registra la factura y se despliega un mensaje de advertencia que informa el ingreso correcto de la factura (Figura 79).

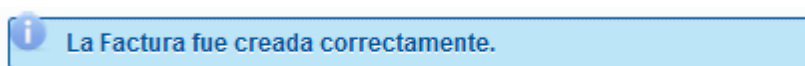


Figura 79 Mensaje de Advertencia Factura

Fuente: El Autor, 2013

5 MÓDULO ASOCIADO

El módulo asociado permite realizar modificaciones en los datos personales así como también generar la capacidad de endeudamiento, retenciones y roles de pago (Figura 80).

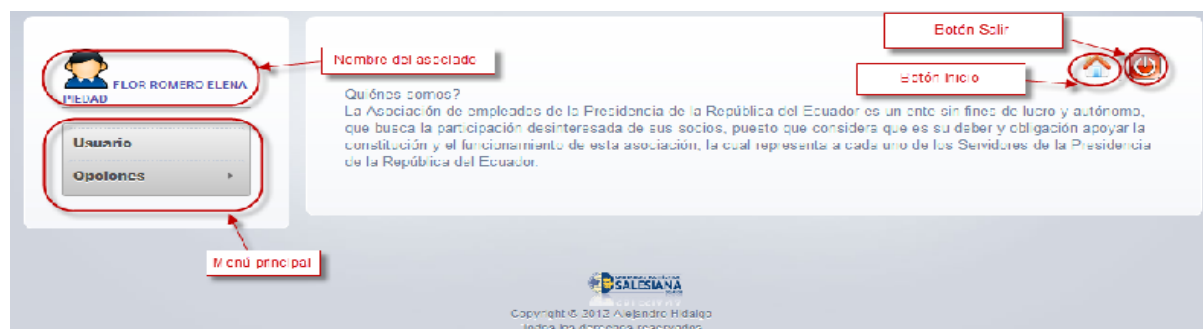


Figura 80 Módulo Asociado

Fuente: El Autor, 2013

5.1 MENÚ USUARIOS

El asociado puede actualizar los datos personales en el menú usuarios.

En la ventana Editar usuario (Figura 81) se puede modificar los campos del usuario, luego haciendo clic en el botón actualizar se despliega un mensaje de información de actualización exitosa (Figura82).

EDITAR USUARIO

Actualizar

Seleccionar botón actualizar usuario

Actualizan do datos

Editar Usuario			
Cédula	0601614324	Apellidos:	FLOR ROMERO
Nombres:	ELENA PIEDAD	Password:	
Dirección:	QUTUMBE	FonoDomicilio:	02658945
Celular:	0990753949	ServicioWap:	<input type="radio"/> NO. <input checked="" type="radio"/> SI.
Nombramiento:	SI	Activo	SI
Sueldo:	0.00	Cargo:	ISTA DE PROYECTOS
CodPerfil:	ASOCIADO		

Figura 81 Editar Campos Usuario
Fuente: El Autor, 2013

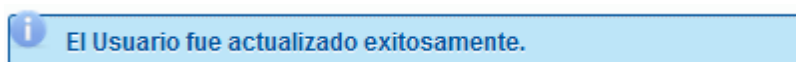


Figura 82 Mensaje de Información Editar Usuario
Fuente: El Autor, 2013

5.2 MENÚ OPCIONES

El menú opciones tiene dos funciones como visualizar los reportes mensuales de las retenciones y de los préstamos.

5.2.1 SUBMENÚ REPORTES MENSUALES

Muestra una lista con todas las retenciones que han sido generadas para ese usuario con una descripción de la misma (Figura83).

REPORTES MENSUALES

Exportar A:

LISTA DE RETENCIONES MENSUALES						
CodRetencion	Cédula	ValorTotal	FechaRetencion	FechaDesde	FechaHasta	columna
1439	0601614324	138.05	19/December/2012	01/December/2012	19/December/2012	

Figura 83 Submenú Reportes Mensuales
Fuente: El Autor, 2013

Arriba de la lista se encuentra el botón exportar con los diferentes tipos de extensiones (Figura84). En el menú de acciones se encuentra en la última columna de la grilla permitirá exportar e imprimir (Figura85) la información mostrada.



Figura 84 Exportar y Formatos
Fuente: El Autor, 2013

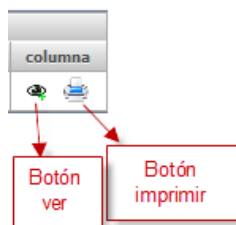


Figura 85 Botones Imprimir y Ver
Fuente: El Autor, 2013

5.2.1.1 RETENCIONES – VER

Después de haber seleccionado el registro que se desea detallar se procede a dar clic en el ícono Ver, y muestra la información en detalle de la retención total para ese mes, (Figura 86). De igual forma se puede imprimir con el ícono correspondiente la lista mostrada.

REPORTES MENSUALES

LISTA DE RETENCIONES TOTALES					
CodRetencion	Cédula	ValorTotal	FechaRetencion	FechaDesde	FechaHasta
55	1212121219	76.00	26/noviembre/2012	01/noviembre/2012	30/noviembre/2012

DETALLE RETENCIONES DE APORTES					
Cédula	Apellido	Nombre	NombreAporte	MontoRetención	FechaRetención
1212121219	DIEZ Y NUEVE		ASOCIACION NO ASOCIADO	10.00	2012-11-25
1212121219	DIEZ Y NUEVE		SISTEMA CELULAR NO ASOCIADO	5.00	2012-11-25
1212121219	DIEZ Y NUEVE		PAGO SMS NO ASOCIADO	1.00	2012-11-25

DETALLE RETENCIONES DE Préstamo					
Cédula	Apellido	Nombre	Préstamo	MontoRetención	FechaRetención
No records found.					

DETALLE RETENCIONES DE CASAS COMERCIALES					
Cédula	Apellido	Nombre	CasaComercial	MontoRetención	FechaRetención
1212121219	DIEZ Y NUEVE		CRVEHOGAR	60.00	2012-11-20

Figura 86 Retenciones Asociado Ver

Fuente: El Autor, 2013

5.2.1.2 RETENCIONES –IMPRIMIR

La opción imprimir muestra una especie de factura con los totales de cada retención y al final la retención total que será debitada. (Figura 87). De igual forma se puede exportar e imprimir.

REPORTES MENSUALES

RETENCIONES MENSUALES DE LA ASOCIACIÓN						
Tipo	Cédula	Apellidos	Nombres	Detalle	Valor	FechaRetención
FACTURA	1212121219	DIEZ Y NUEVE		CRVEHOGAR	60.00	2012-11-20
APORTE	1212121219	DIEZ Y NUEVE		ASOCIACION NO ASOCIADO	10.00	2012-11-25
APORTE	1212121219	DIEZ Y NUEVE		SISTEMA CELULAR NO ASOCIADO	5.00	2012-11-25
APORTE	1212121219	DIEZ Y NUEVE		PAGO SMS NO ASOCIADO	1.00	2012-11-25
					\$ 76.00	
					ValorTotal \$76.00	

Figura 87 Retenciones Asociado Imprimir

Fuente: El Autor, 2013

5.2.2 SUBMENÚ PRÉSTAMOS

Permite visualizar los préstamos activos que tiene el asociado con información detallada. Arriba de la lista se encuentra el botón imprimir, dando clic se puede imprimir la lista de los préstamos (Figura 88)

REPORTE DE PRESTAMOS



TABLA LISTAR PRÉSTAMO					
CodPrestamo	Asociado	ValorCheque	Cuotas	FechaPréstamo	CuotasRestantes
PRE_0002	DIEZ Y NUEVE	400.00	4	20/12/2012	3

Figura 88Submenú Préstamos
Fuente: El Autor, 2013

5.2.3 SUBMENÚ CAPACIDAD DE ENDEUDAMIENTO

La capacidad de endeudamiento es la opción que nos permite ver si un asociado cumple con los requisitos establecidos por la asociación para que se pueda otorgar un préstamo, ser garante del mismo o saber si puede endeudarse conociendo los pagos mensuales que le serán debitados. Para esto se tiene como antecedentes los siguientes puntos:

- El asociado debe pertenecer más de un mes en la asociación.
- El valor de gastos de la presidencia debe haber sido ingresado.
- El sueldo del asociado debe haber sido ingresado.
- El valor de sueldo y gastos de la presidencia deben ser mayores a 0.

Para mayor detalle ver el ítem 3.2.8 3.2.8 Submenú Capacidad de Endeudamiento.

5.2.4 SUBMENÚ ROL DE PAGOS

La opción de Rol de Pagos nos detalla el valor que será depositado en la cuenta de cada asociado, esto después de una serie de cálculos que respectan entre algunos al envío de los gastos externos por parte de la Presidencia de la

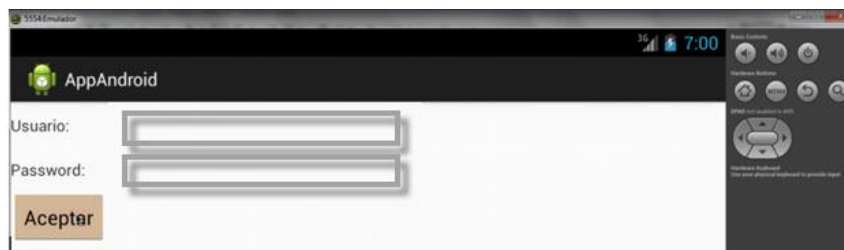


Figura 90 Interfaz Android
Fuente: El Autor, 2013

Al llenar los datos correctos se debe proceder dar clic en el botón Aceptar, para que pueda acceder, mostrando el asociado que ingresó, y con el detalle, en este menú se debe presionar cualquiera de las dos opciones (Figura 91).

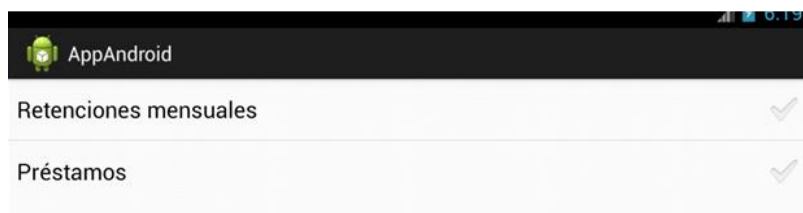


Figura 91 Interfaz Android 2
Fuente: El Autor, 2013

6.1 RETENCIONES MENSUALES

Al presionar en esta opción detalla el total de cada retención generada para el asociado que ingresó previamente al sistema, con la fecha del mes que se efectuó la retención (Figura 92).

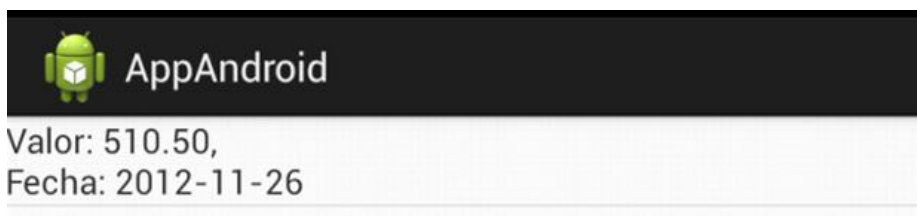


Figura 92 Retenciones Mensuales
Fuente: El Autor, 2013

Al presionar en esta fecha va a una siguiente pantalla donde detalla las retenciones para llegar al valor seleccionado (Figura93).



Nombre: ASOCIACION ASOCIADO
Valor: 5.00
Fecha: 2012-11-26

Nombre: PAGO WEB
Valor: 3.50
Fecha: 2012-11-26


Nombre: SEMPPO
Valor: 400.00
Fecha: 2012-11-26

Retención Préstamo: 1
Monto: 102.00

Figura 93 Detalle Retención Total
Fuente: El Autor, 2013

6.2 PRÉSTAMOS

Al presionar en esta Opción detalla el valor del préstamo y las cuotas que se van cancelando (Figura 94).



Cuotas: 3
Cuotas restantes: 2
Fecha préstamo: 2012-11-26

Figura 94 Detalle Préstamo
Fuente: El Autor, 2013

ANEXO C. MANUAL TÉCNICO

1 INTRODUCCIÓN.

El sistema desarrollado AsoPresi es un software que se maneja en ambiente Web, y es por esta razón que se requiere dar a conocer los requerimientos necesarios para que se pueda levantar el servidor sin ningún inconveniente en cualquier otro servidor de aplicaciones o en un futuro tener las pautas necesarias para poder migrar a nuevos requerimientos de sistemas operativos.

El sistema al usar herramientas de software libre como fue su objetivo en un principio es capaz de adecuarse a los diversos tipos de sistemas operativos que trabajen con los requerimientos mencionados posteriormente.

2 REQUERIMIENTOS

2.1 REQUERIMIENTOS DE HARDWARE

En cuanto a requerimientos de hardware se requiere que sea un servidor que funcione 24 horas los 7 días de la semana, con las siguientes características mínimas.

- Procesador: Intel(R) Pentium(R) III CPU 1.26 Ghz
- Memoria RAM: 1.00 GB
- Disco Duro: 128 GB
- Unidad DVD-RW
- Tarjeta de Red

2.2 REQUERIMIENTOS DE SOFTWARE

La aplicación al ser desarrollada en Java requiere de los siguientes componentes para su utilización, las cuales se especifican con las versiones que se efectuó la implementación.

- Sistema operativo (Preferible Software Libre CentOS 6; Windows Server 2003 o 2008)
- Jdk-7u7
- Glassfish-3.0.1
- PostgreSQL-9.2.1
- Jboss-as-7.1.1.Final

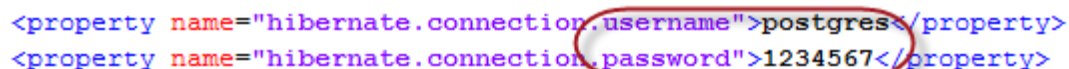
3 IMPLEMENTACIÓN

A continuación se establecen los pasos a seguir y requerimientos necesarios para que se pueda levantar la aplicación sin ningún inconveniente.

3.1 BASE DE DATOS - postgres

La base de datos que se va a usar es postgres, preferiblemente la última versión estable, después de tener instalado postgres se debe configurar la base de datos, y el servidor debe obligatoriamente tener la misma clave de acceso que la aplicación web configurada.

En la aplicación el archivo de configuración se especifica el usuario y contraseña del gestor de base de datos que se encuentra en la siguiente ruta: ...\\AsoPresi\\src\\java\\hibernate.cfg.xml, además del puerto: 5432 mostrado en el mismo archivo.



```
<property name="hibernate.connection.username">postgres</property>
<property name="hibernate.connection.password">1234567</property>
```

Figura 1: Archivo de Configuración en la Aplicación
Fuente: El Autor, 2013

Esta información deberá ser la misma al configurar en el gestor de base de datos, como se aprecia en la FIGURA 2, el usuario es postgres al igual que la FIGURA 1, y la contraseña o password deberá ser la misma en este caso: 1234567

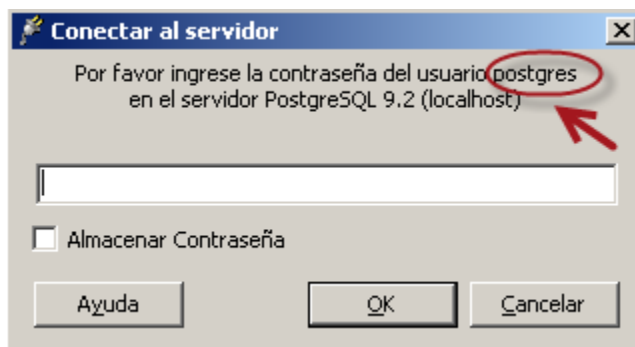


Figura 2: Configuración en Postgres.
Fuente: El Autor, 2013

3.2 RESTAURAR BDD

El siguiente paso es restaurar la base de datos, para esto se debe tener el archivo de back up o respaldo y ubicarlo en una ruta para importar los datos.

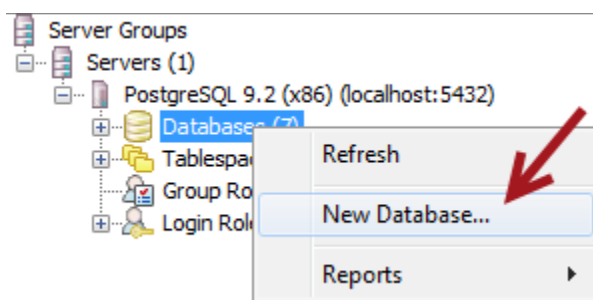


Figura 3: Nueva Base de Datos.
Fuente: El Autor, 2013

Ingresa el nombre de la nueva base de datos tal como se muestra en la FIGURA 4: bd_asopresi

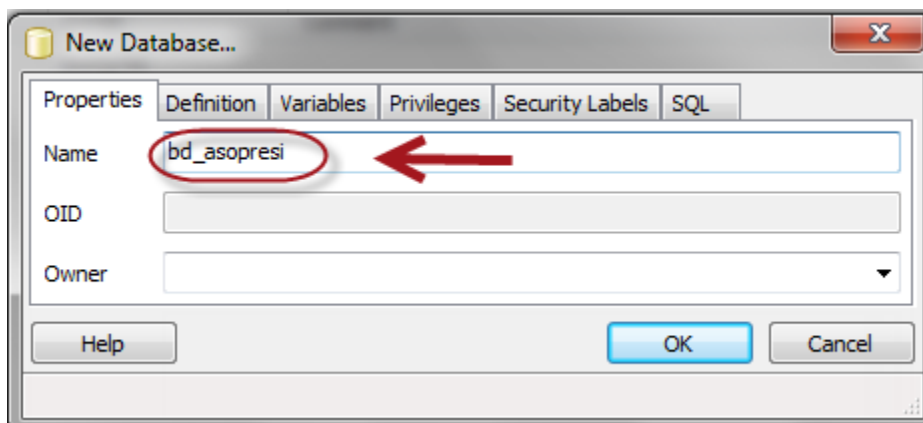


Figura 4: Nombre de Base de Datos.
Fuente: El Autor, 2013

Después de haber creado la base el siguiente paso es restaurar desde un archivo, para esto se debe dar clic derecho en la base de datos creada y presionar Restore o Restaurar.

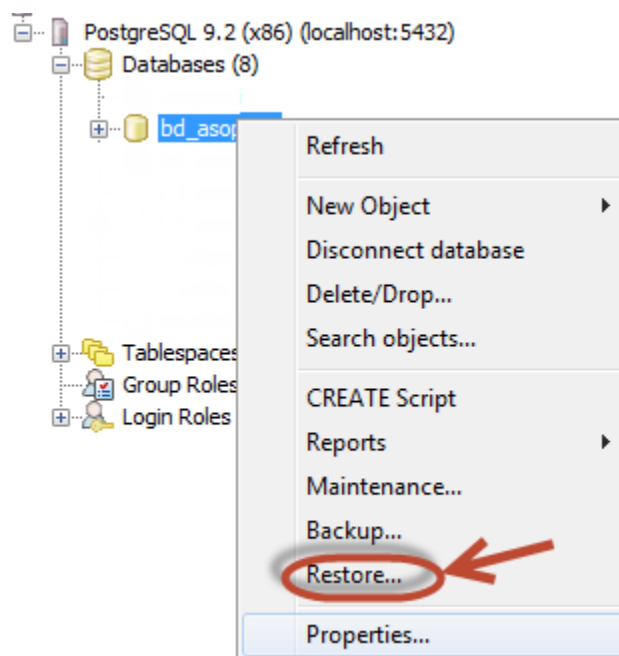


Figura 5: Pasos Restaurar Base de Datos.
Fuente: El Autor, 2013

Seleccionar el archivo, escoger como Rolename: postgres y dar clic en Restore o Restaurar.

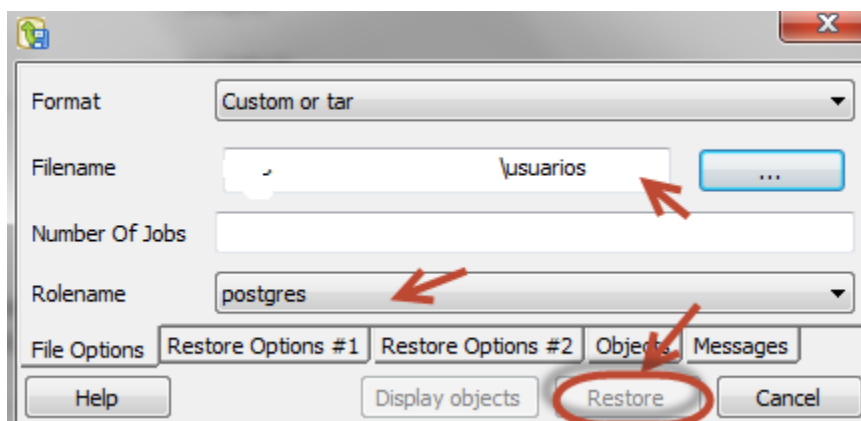


Figura 6: Restaurar Base de Datos.
Fuente: El Autor, 2013

Si el proceso retorna: **Processreturnedexitcode 0**, quiere decir que se restauró la base de datos exitosamente.

3.3 LEVANTAMIENTO DE APLICACIÓN WEB

Después de tener instalado el servidor de aplicaciones Glassfish, ubicarse en la ruta donde se instaló y acceder a la siguiente ruta: ...\\glassfish-3.1.2.2\\glassfish\\bin\\ y ejecutar como administrador el siguiente archivo (clic derecho ejecutar como administrador): asadmin.bat.

Escribir el siguiente comando dentro de la línea de comando que aparece: start-domain domain1 (domain1 es el dominio donde se está trabajando) y presionar enter., como se indica en la FIGURA 7.



```
Use "exit" to exit and "help" for online help.
asadmin> start-domain domain1
```

Figura 7: Nueva Base de Datos.
Fuente: El Autor, 2013

Abrir un navegador o browser como: Mozilla o Internet Explorer y llamar a la siguiente dirección: <http://localhost:4848/> y escribir el usuario y contraseña para poder ingresar.



Figura 8: Abrir Glassfish
Fuente: El Autor, 2013

Ubicarse en la pestaña Aplicación y seleccionar Desplegar.



Figura 9: Aplicaciones Glassfish
Fuente: El Autor, 2013

Seleccionar el archivo .war de AsoPresi, que es el un archivo empaquetado con toda la información necesaria para levantar la aplicación, y proceder a dar clic en el botón Aceptar.

Desplegar Aplicaciones o Módulos

Especifique la locación de la aplicación o módulo a desplegar. Una aplicación puede estar en un



* Indica que es un campo

Ubicación: ☒ Archivo empaquetado que se cargará en el servidor

Figura 10: Desplegar la Aplicación
Fuente: El Autor, 2013

Como se muestra en la FIGURA 11, la aplicación ha sido cargada exitosamente y podemos dar clic en Iniciar para que la ventana principal aparezca.



Figura 11: Iniciar la Aplicación
Fuente: El Autor, 2013



Figura 12: Aplicación Levantada
Fuente: El Autor, 2013

3.4 LEVANTAR LA APLICACIÓN ANDROID

Ubicarse donde se haya copiado la carpeta de Jboss en el servidor, e ir a la siguiente ruta: ... \jboss-as-7.1.1.Final\bin

Abrir un terminal y ubicarnos en la dirección mencionada, escribiendo el comando: standalone.bat -b 172.17.42.10 (Esta es la ip con la que cuenta el servidor) posteriormente dar enter.

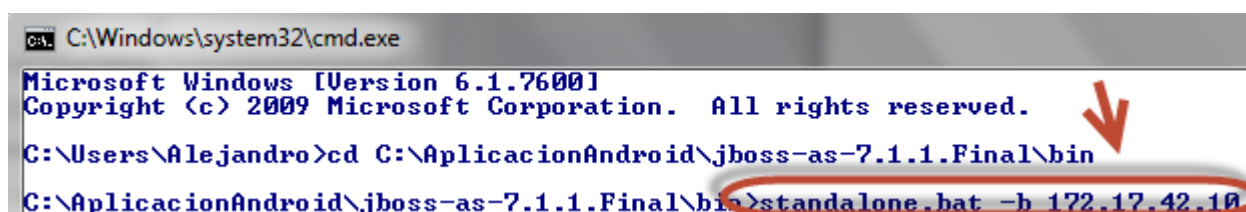
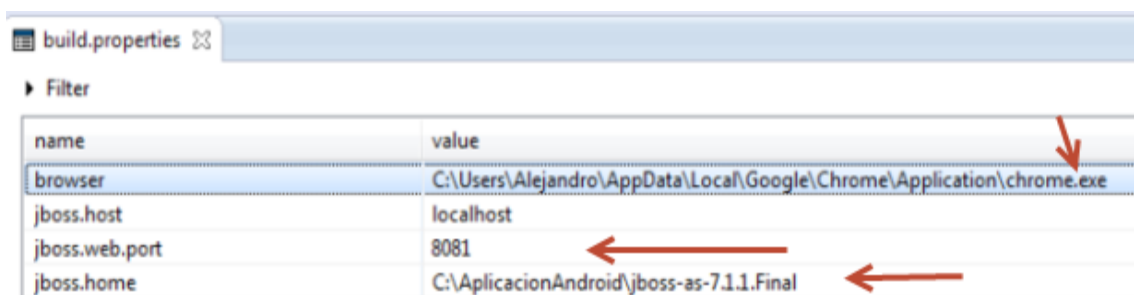


Figura 13: Levantar Jboss
Fuente: El Autor, 2013

De igual forma verificar que los datos de la configuración de archivos sean los mismos para el servidor de aplicación, en el archivo de la aplicación build.propertiesDescuentosWs se encuentra donde se ubica el servidor Jboss y el puerto en el cual trabaja.



name	value
browser	C:\Users\Alejandro\AppData\Local\Google\Chrome\Application\chrome.exe
jboss.host	localhost
jboss.web.port	8081
jboss.home	C:\AplicacionAndroid\jboss-as-7.1.1.Final

Figura 14: Configuración Previa
Fuente: El Autor, 2013

Con estos antecedentes nos podremos ubicar en el navegador escribiendo la siguiente ruta: <http://localhost:8081>, introduciendo el usuario y clave respectivo, como se muestra en la FIGURA 15.

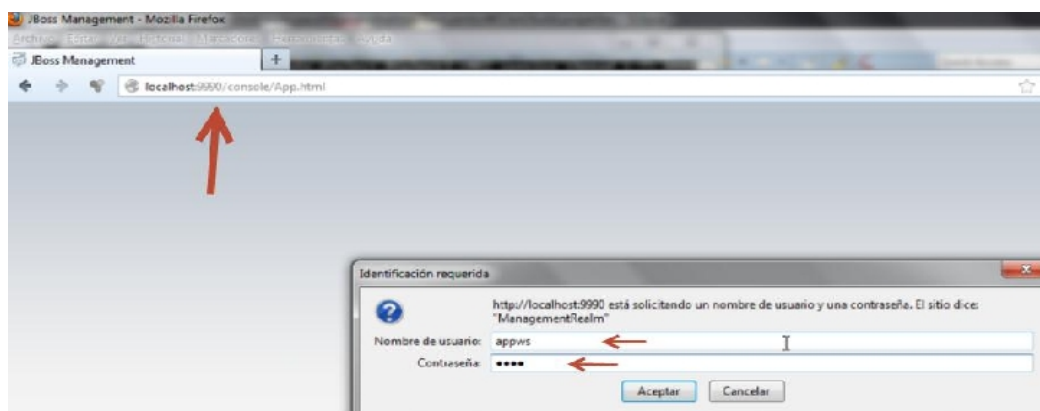


Figura 15: Configuración Previa Jboss
Fuente: El Autor, 2013

Para cargar la aplicación nos Ubicamos en el menú de la parte derecha y seleccionamos ManageDeployments y luego en Add Content

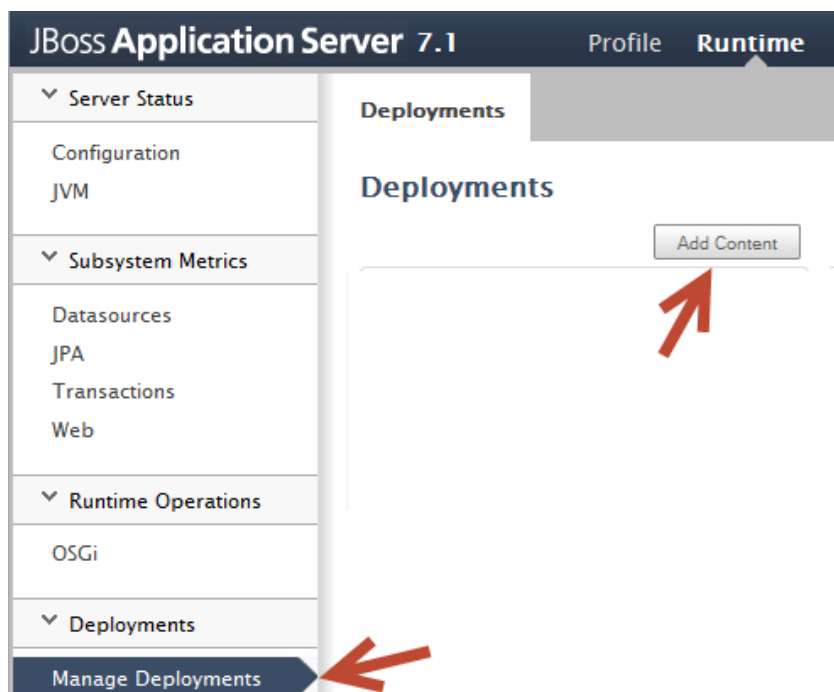


Figura 16: Desplegar Aplicación Jboss
Fuente: El Autor, 2013

Esta es la configuración básica para que las aplicaciones puedan trabajar sin ningún problema.